

Developing and Debugging OS/2 LAN Applications

ANNUAL LAN ISSUE
INCLUDING: **BUYER'S GUIDE**
LAN DEVELOPMENT TOOLS

March/April 1994

Display Until 4/30/94

OS/2 Developer

THE MAGAZINE FOR ADVANCED SOFTWARE DEVELOPMENT

**DEVELOPING
OS/2 LAN
APPLICATIONS**

**Buyer's Guide:
LAN Development Tools**

LAN NetView Development

**Network Management
Using SNMP**

**Debugging NetBIOS
Applications**

**SUPER
NETWORK
DEVELOPMENT
ON
OS/2**

U.S. \$9.95 Vol. 6 No. 2



A Miller Freeman Publication

WATCOM™

Visual Development Environment For

WATCOM VX•REXX is an easy to use visual development environment for creating applications that leverage the capabilities of OS/2 2.x and exploit the Presentation Manager graphical user interface. VX•REXX combines a project management facility, visual designer and an interactive source-level debugger to deliver a very approachable and highly productive visual development environment.

Design Applications Visually

Create rich graphical applications quickly and easily using the visual design environment. With the visual designer, you can graphically create Presentation Manager interface objects, quickly customize their properties, and easily attach REXX procedures using powerful drag-and-drop programming techniques.

Integrated Development Environment

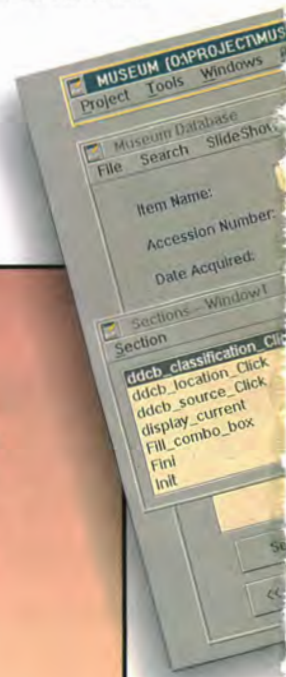
Build, test and debug your application without leaving the development environment. Then package your application as an EXE file or PM macro for royalty-free redistribution. The power of the integrated development environment and debugger can also be used with your existing REXX applications.

Powerful Open Environment

Enjoy the simplicity of event-driven programming together with the global editing capabilities essential for professional project management. WATCOM VX•REXX is open and extensible through IBM's object oriented System Object Model (SOM) technology. You can access all standard REXX API's including DB2/2, because VX•REXX is based on the OS/2 2.x standard system REXX.

Highlights

- ▶ Easy to use visual development environment
- ▶ Create and modify objects dynamically at both edit and run time
- ▶ Powerful project management facility
- ▶ Advanced interactive source-level debugger
- ▶ Package your applications as EXE files or PM macros
- ▶ Access to standard REXX API's including DB2/2
- ▶ System Object Model (SOM) based object manager
- ▶ Drag-and-drop programming
- ▶ Support for multi-threaded applications
- ▶ Include OS/2 style help and hints in your applications
- ▶ Supports SAA CUA'91 objects
- ▶ Integrated console window support for existing REXX programs
- ▶ Royalty-free run-time
- ▶ Multiple modeless window support
- ▶ Create PM macros for applications supporting REXX as a macro language



VX•REXX™

OS/2 REXX

Interactive Debugging

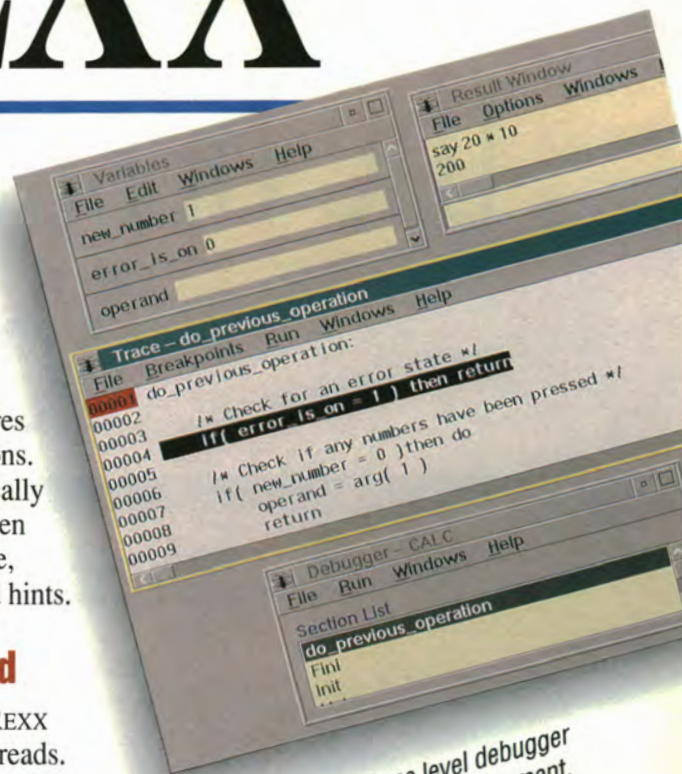
If an error occurs at run-time, VX•REXX will display a traceback pinpointing the source line where the error occurred. A simple click of the mouse will return you to the source edit window to correct the error. The built-in interactive source-level debugger lets you set breakpoints, step through code and watch variables to track down complex problems.

Build Professional Applications

WATCOM VX•REXX allows you to leverage key OS/2 features to create professional applications. Build applications that dynamically create and modify CUA'91 screen objects at both edit and run-time, and include OS/2 style help and hints.

Create Multi-Threaded Applications

Every VX•REXX application contains multiple threads. One thread remains responsive to user input while others continue processing. In addition, VX•REXX provides the ability for advanced applications to easily use additional threads.



The integrated source level debugger simplifies your project development.



WATCOM VX•REXX:

The integrated visual solution builder for OS/2 2.x.

Suggested Retail: \$199*

**Call Toll Free
1-800-265-4555**

WATCOM

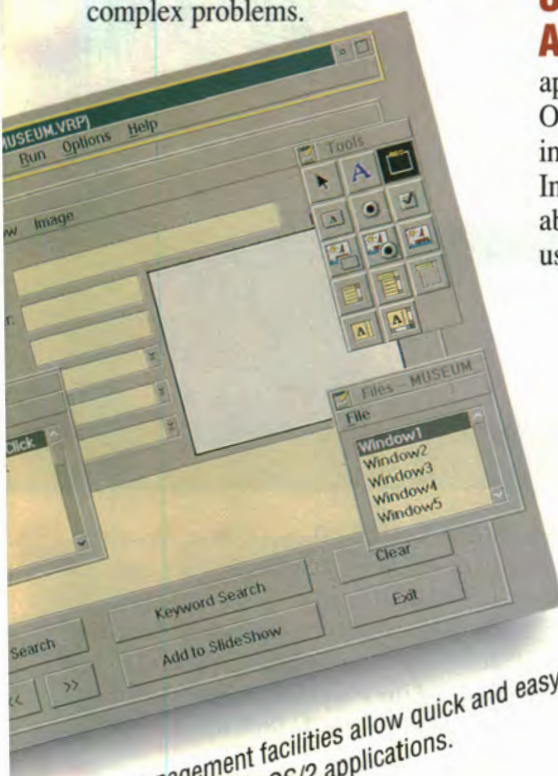
WATCOM International

415 Phillip Street, Waterloo, Ontario, Canada, N2L 3X2
Phone: (519) 886-3700 Fax: (519) 747-4971

*Prices and specification are subject to change without notice. Price does not include freight and taxes where applicable. Prices quoted in US dollars. WATCOM, the Lightning Device, and VX•REXX are trademarks of WATCOM International Corporation. Other trademarks are the properties of their respective owners. © Copyright 1993 WATCOM International Corporation.

Circle Reader Service Number 1

Project management facilities allow quick and easy development of your OS/2 applications.



We didn't say COBOL Workbench is the best in the world. You did.

 PC SOFTWARE BRAND PREFERENCE		DOS-Based CASE		AMONG TOTAL RESPONDENTS Question: Which brands of DOS-based CASE software are currently installed in your organization?	
INSTALLED IN COMPANY		EASY TO USE		BEST TECHNOLOGY	
KnowledgeWare, Inc. (Information Engineering Workbench)	42%	MicroFocus (Workbench)	38%	MicroFocus (Workbench)	29%
MicroFocus (Workbench)	40%	KnowledgeWare, Inc. (Information Engineering Workbench)	17%	Texas Instruments (Information Engineering Facility (IEF))	26%
Intersolv (Excellerator)	15%	Intersolv (Excellerator)	9%	KnowledgeWare, Inc. (Information Engineering Workbench)	24%
Texas Instruments (Information Engineering Facility (IEF))	13%	Computer Associates (CA-REALIA)	8%	Computer Associates (CA-REALIA)	5%
Computer Associates (CA-Tellon)	7%	Visible Systems Corp. (Visible Analyst Workbench)	8%	Cadre Technologies, Inc. (Termux)	3%
BEST SERVICE/SUPPORT		BEST DOCUMENTATION		PREFER TO DO BUSINESS WITH	
KnowledgeWare, Inc. (Information Engineering Workbench)	26%	MicroFocus (Workbench)	34%	MicroFocus (Workbench)	28%
MicroFocus (Workbench)	26%	KnowledgeWare, Inc. (Information Engineering Workbench)	22%	KnowledgeWare, Inc. (Information Engineering Workbench)	22%
PLAN TO BUY					
MicroFocus (Workbench)	50%				
KnowledgeWare, Inc. (Information Engineering Workbench)	29%				

We've always believed Micro Focus COBOL Workbench® is the best in the world. Now you've confirmed it by unequivocally placing it first in the 1993 Computerworld PC Software Brand Preference survey.

You voted COBOL Workbench as **Best Technology** and **Easy to Use**. That's not all. Workbench has also won top honors in **Price/Performance**, **Best Documentation**, **Plan to Buy**, and most importantly, you chose Micro Focus as the company you **Prefer to do Business With**. According to you there was no contest.

Not surprising really. There is no better technology for developing new systems or re-engineering existing applications on the

workstation. Programmers find Workbench puts them directly in control of their development environment, delivering quality business applications on time and on budget.

If it isn't Micro Focus COBOL Workbench, it isn't in the running. That's not just our opinion, it's yours.

For your free copy of the 1993 Computerworld survey, or for more information, call 800-MF-COBOL, (800-872-6265).

MICRO FOCUS®

Micro Focus Inc. 2465 East Bayshore Road, Palo Alto, CA 94303. Tel. (415) 856 4161.

Circle Reader Service Number 2

Micro Focus and COBOL Workbench are registered trademarks of Micro Focus, Inc. All other trademarks are property of their respective companies.

GSA Contract Number GS00K93AGS6403. In Canada call IBM Canada at 1-800-465-1234.

OS/2 Developer

MARCH/APRIL 1994

THE MAGAZINE FOR ADVANCED SOFTWARE DEVELOPMENT



EDITOR'S COMMENTS

The Decade of the LAN 5



PROGRAMMING INSIDER

Efficient 32-bit Memory Management With OS/2 6



CORPORATE STUDY

Insurance Company Looks to Future With OS/2 12



LANS

Locking Network File System Files on OS/2 18

Using the LAN NetView Basic User Interface APIs 29

Network Management: Using SNMP for LAN NetView 46

NetBIOS and 802.2 Applications: Problems and Solutions 55

Enabling Your Applications for CID 66

Increasing LAN Efficiency With CM/2 Distributed Feature 72

LAN Development Tools Buyer's Guide 89



PRODUCT WATCH

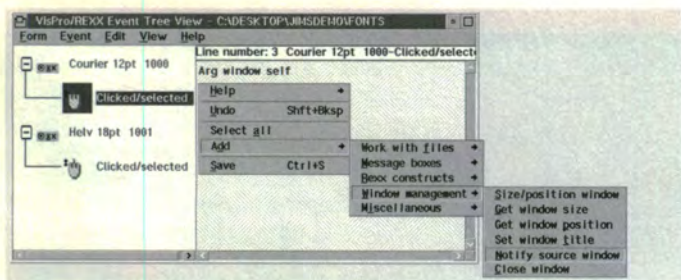
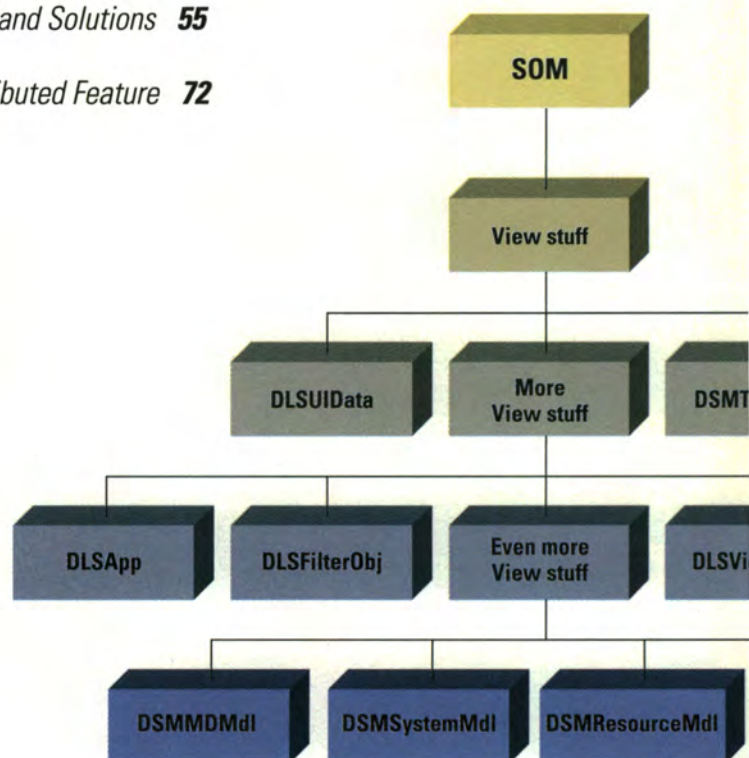
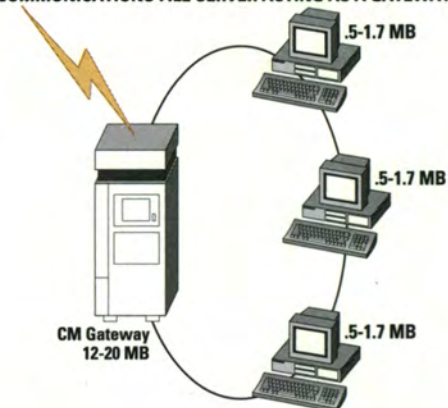
New Tools and Utilities 78



DATABASE

Visually Sampling DB2/2 81

COMMUNICATIONS FILE SERVER ACTING AS A GATEWAY





Programmer's Paradise®

Call for a
FREE
Catalog!



SourceLink v2.0 by One Up Corporation

SourceLink, the ultimate 32-bit programming development tool, combines the functionality of hyperlink source code access, a fully-functional editor and extensive file manipulation utilities into one very powerful toolset. Now you can find code, change code, spawn compiles and hyperlink directly to the source of error quickly and efficiently. Features point & click source code navigation and automatic generation of function call trees.



List: \$299 Ours: \$259 FAXcetera #: 6005-0003

CA-REALIZER by Computer Associates

Defines a new generation of development tools that handles the mechanics of event-driven programming, message passing, process sharing and other complexities behind the scenes. Combines a structured superset of BASIC extended to access Windows and OS/2 objects and resources. CA-Realizer will help you create spreadsheets, charts, text editors, animation, graphics tablets and user-friendly forms from tools that can be created and manipulated by simple commands.



List: \$295 Ours: \$79 FAXcetera #: 1004-0008



Window Washer v2.0 by One Up Corporation

The latest version of the best-selling 32-bit screen saver for OS/2, with full system password security and the most complete monitor burn-in protection available today. Version 2.0 features many exciting animated effects, digital video and audio (plays CD's, MIDI or WAV files with program effects). Also utilizes TIF, GIF, BMP, & PCX backgrounds.

List: \$40 Ours: \$35 FAXcetera #: 6005-0001



WATCOM VX•REXX by WATCOM

WATCOM VX•REXX Version 2.0 is an easy to use visual development environment for creating applications that leverage the capabilities of OS/2 2.x and exploit its CUA '91 graphical user interface. VX•REXX combines a project management facility, visual designer and an interactive source-level debugger to deliver a very approachable and highly productive visual development environment. With the visual designer, you can create an attractive graphical interface using VX•REXX's full suite of CUA '91 interface objects, quickly customize their properties, and easily attach REXX procedures to the objects. Drag-and-drop programming lets you write event routines without typing any code.

List: \$199 Ours: CALL FAXcetera #: 1683-0016

Open Shutter v1.11 by One Up Corporation

Our easy-to-use screen capture process allows you to select any rectangular area, window, or entire desktop, and capture with a single user-defined keystroke or mouse click. Rotate, change colors, stretch/compress and then preview your modifications. Output to printer, clipboard, or soft copy in a variety of formats (BMP, ICO, TIFF, GIF, IMG, metafile, MacPaint).



List: \$70 Ours: \$59 FAXcetera #: 6005-0002

Ami Pro 3.0 for OS/2 by Lotus Development Corp.

OS/2 users now have a native, 32-bit version of the award-winning Ami Pro 3.0 for Windows. Written from the ground up for OS/2, the new version of Ami Pro provides the highest level of word processing, networking and development power among OS/2 word processors.



Upgrade: List: \$495 Ours: \$369
List: \$129 Ours: \$ 99
FAXcetera #: 4000-0011

GpfRexx by Gpf Systems, Inc.

WYSIWYG visual OS/2 PM programming with REXX. Point and click to create PM design using the full CUA '91 control set (e.g., Notebooks, Containers, etc.), as well as Multi-Media Controls. GpfRexx supports standard OS/2 features like Drag and Drop and Multi-Tasking as well as the extended features, including: Multi-Media, SQL (DB2/2 including Remote services), and advanced communications (APPC, CPI-C, EHLLAPI). No royalties.



List: \$247 Ours: \$149 FAXcetera #: 3582-0005

Special through March 31, 1994!

Circle Reader Service Number 3

GUARANTEED BEST PRICES!(Call for details)

To order call: 800-445-7899

Corporate (CORSOFT): 800 422-6507

FAX: 908 389-9227

International: 908 389-9228

Customer Service: 389-9229

Programmer's Paradise Italia:

39-2-480-16053

For more information on the products featured on this page call—

FAXcetera®: (201) 762-1378

Programmer's Paradise®

1163 Shrewsbury Avenue
Shrewsbury, NJ 07702

• All prices are subject to change without notice.
• Call for details on return policy and shipping charges.



EDITOR

Dick Conklin

EXECUTIVE EDITOR

Nicole Freeman

MANAGING EDITOR

Kelly A. Leighton

EDITORIAL ASSISTANT

Andrea Pucky

EDITORIAL ARTIST

Peter Altenberg

GROUP DIRECTOR

Don Pazour

PUBLISHER

Cathy Passage

ADVERTISING SALES

Angela Barnett

(415) 905-4983

Holly Meintzer

(212) 626-2275

Michele Blake

(212) 626-2322

Dave Moreau

(212) 626-2318

Patricia Sutton

(212) 626-2498

MARKETING MANAGER

Susan McDonald

ART DIRECTOR/MARKETING

Christopher H. Clarke

ADVERTISING PRODUCTION**COORDINATOR**

Tom Marzella

DIRECTOR OF PRODUCTION

Andy Mickus

DIRECTOR OF CIRCULATION

Jerry Okabe

CIRCULATION DIRECTOR

John Rockwell

CIRCULATION MANAGER

Lucinda Formyduval

SUBSCRIPTION INQUIRIES

U.S.: (800) WANT-OS2

Foreign: (708) 647-5960



BY DICK CONKLIN

Editor's Comments

The Decade of the LAN

I finally received my long-awaited ThinkPad 500 and wasted no time installing OS/2. This little number is IBM's smallest computer to date, their first in the sub-notebook class.

I've always been amazed at the power of the smallest machines, ever since I attempted to earn a living by selling the infamous IBM model 5100 Portable Computer in my Dade County, Florida sales territory. After lugging that 55-pound sucker around downtown Miami for two years, I resolved that someday I would own a computer that was *really* portable.

But, predicting smaller computers back in 1976 didn't require a crystal ball. We all knew that someday these computers were going to shrink in size, have more memory, and run faster—for less money. Displays would have high-resolution color screens. Who knows, asynch communication speeds might exceed 110 baud. It was just a matter of time.

EMBRACING THE FUTURE

Most of us accepted that mainframe computers would get much smaller and desktop systems would get more powerful and that someday, the two would meet. But few prognosticators in the 1970s foresaw the local area networks of today. The first LANs exploited shared resources—file and print servers. Then came office e-mail. Then database and application servers. And client/server applications.

OS/2 Developer's LAN section has always been popular with readers. As corporate America embraced OS/2, it also accepted it as a client/server development platform, requiring a whole new set of programming skills. In this issue, we feature several articles on LAN application development. We deal with shared files, LAN application distribution and installation, communicating beyond the LAN, debugging a LAN application, and more on NetView



Dick Conklin

Contact	CompuServe	Internet
Dick Conklin (Editorial)	76711,1005, or OS2DF2 Forum	os2mag @vnet.ibm.com
Miller Freeman (Circulation)	71572,341	71572,341 @compuserve.com
Cathy Passage (Publisher)	71673,1163	71673,1163 @compuserve.com
Kelly Leighton (Managing Editor)	73632,2652	73632,2652 @compuserve.com

APIs. In our Buyers Guide, you'll find a mini-directory of several OS/2 development tools for building LAN-exploiting applications.

Please note that Compuware's correct contact information is: Phone: (313) 737-7300, Fax: (313) 737-7108. Their product, RemoteControl/2, was listed in the Desktop DB2 supplement's buyers' guide.

OS/2 Developer: Vol. 6, No. 2, March/April 1994

Subscription information: U.S.: One year (six issues), \$39.95. Canada, Mexico, and international surface mail, add \$16 per year for postage. Canadian GST no. 124513185. International air mail, add \$30 per year for postage. Foreign orders must be accompanied by payments in U.S. funds. For new orders and customer service, call (800) WANT-OS2 (800-926-8672) or (708) 647-5960 (fax: 708-647-0537). IBM employees and branch office customers can subscribe to OS/2 Developer through IBM Mechanicsburg's Systems Library Services (SLSS) using OS/2 Developer's order number: G362-0001. POSTMASTER: Please send address changes to OS/2 Developer, P.O. Box 1079, Skokie, IL 60076-8079. Allow eight weeks for subscriptions to begin. OS/2 Developer (ISSN 1073-0729) is published bimonthly by Miller Freeman Inc., 600 Harrison Street, San Francisco, CA 94107, (415) 905-2200. Application to mail at second class postage rates is pending at San Francisco, CA and additional mailing offices.

OS/2 Developer has applied for BPA membership.

© Copyright 1994 by Miller Freeman Inc. PRINTED IN THE U.S.A.



Programming Insider

In this issue's *Programming Insider* column, we describe what goes on under the covers of 32-bit memory management and show you how to apply some powerful OS/2 concepts. **By DAVID REICH**

Efficient 32-bit Memory Management With OS/2



David Reich

Now that software is up to speed with 32-bit hardware, programmers are challenged to manage system resources efficiently. In the relatively new 32-bit multitasking world, your program is no longer the only one running in the system. In fact, you have awesome power over what you can do with your programs and the entire system.

The flat memory model frees you of the segmented memory and pointer arithmetic you once used when working with large memory objects. However, using this new model presents hurdles and requires new ways of thinking.

NO MORE SEGMENTS

The first thing you will notice when comparing OS/2's 32-bit memory management with prior systems—such as DOS—is that there are no more segments. From the program's perspective, a memory object is the basic unit of allocation. Memory objects can be anywhere from one byte to 512MB in size. (The current implementation of OS/2 limits each process or program to 512MB.) An object is physically allocated in 4K page increments. This means that a small object between one and 4K in size will still occupy one physical page in memory. A 4K plus 1

through 8K object will occupy two pages and so on.

Memory is paged out using a modified least-recently-used (LRU) algorithm. Changed pages are selected for disk swapping based on when they were last used. Whole memory objects are not paged at once; they are paged only one 4K page at a time, as necessary.

16-bit Selectors, Offsets, and Descriptors. Let's review the terminology we use in a 16-bit segmented memory model. When memory is allocated and accessed, it is addressed with a 16-bit selector, or segment, and a 16-bit offset (represented as 16:16). The selector points to a descriptor in a table, which describes the physical segment. The offset then specifies where in the segment the required bytes are located.

Since segments are not uniform in size, a new segment allocated or swapped back into memory may not find an available space of its exact size. Holes, which are pockets of unused memory, tend to accumulate.

NOW-LINEAR ADDRESSES

In the 32-bit scheme, addresses are represented as 0:32. That is, they can be viewed as a 32-bit flat offset into a huge linear address space. Linear space is mapped by a linear, or flat, 0:32 address.



Actually, memory addresses are implemented as 16:32, where one selector maps the entire system address space, but the details of that mechanism are of no concern to applications and are beyond the scope of this discussion. Under the covers, however, mapping is taking place.

This 32-bit offset is really an ordered tuplet, as shown in Figure 1, representing a page table, a page frame within the table (these two are analogous to the 16-bit selector and descriptor), and an offset within the page frame. Using these tables and the offset, each linear address maps to a unique memory location, as shown in Figure 2.

The tables abstract the memory locations so the OS/2 memory manager can move pages in and out of physical memory transparently. In this paged environment, no further movement is needed, since all holes in memory are exactly the same size. The result is better application and system performance.

PAGES AND ATTRIBUTES

Each page has attributes. One that you cannot directly control is whether it is present in memory or paged out. This is a function of the OS/2 memory manager. You will soon see how you can maximize the amount of your application that is physically present in memory, as compared to paged out, even in low-memory situations.

Each page of a memory object has its own unique set of attributes. The attributes you can directly control include **READ ONLY**, **READ/WRITE**, **EXECUTABLE**, **COMMITTED**, and **GUARD**. The first three are self-explanatory; **EXECUTABLE** pages are also inherently **READ ONLY**. The **COMMITTED** and **GUARD** page attributes are of special interest in optimizing memory usage.

OS/2 memory objects can be allocated as committed, which means that all pages referenced by the set of addresses returned from **DosAllocMem** are physically available. Uncommitted memory has linear address space reserved, but those page frames have no pointers mapped to physical memory. If your program attempts to use this memory before you commit it, a protection exception is raised and the process terminates. An application can commit previously uncommitted memory either by calling **DosSetMem** to change its attributes or by using guard pages.

A guard page is a special page of uncommitted memory. When an address that maps to a guard page (one with the **GUARD** attribute) is touched, a guard page exception—as opposed to a page-not-present exception—is raised. The OS/2 system default exception handler for guard page exceptions does two things, as a result. First, it commits the page. Next, it sets the guard page attribute for the next page in the memory object. If there are no

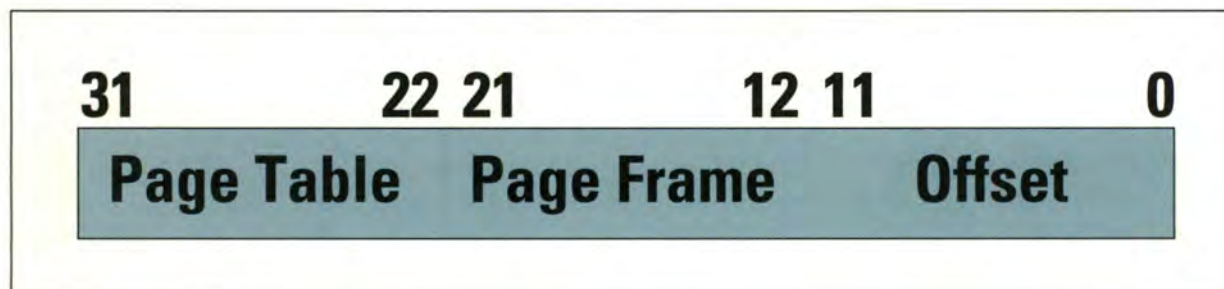


Figure 1. Structure of an OS/2 linear address

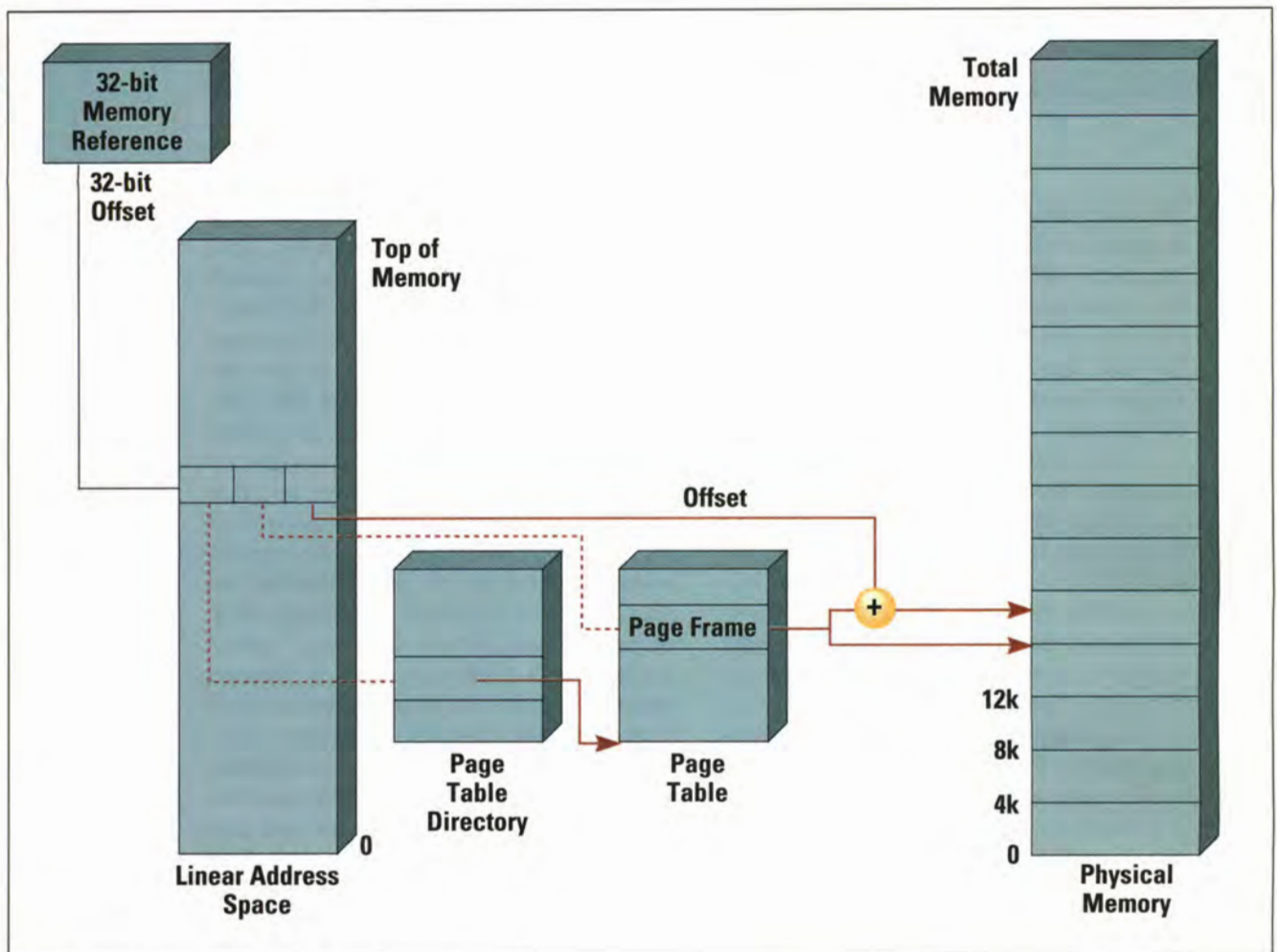


Figure 2. OS/2 linear address to physical memory mapping

more pages, the flag `Cannot Grow Stack` is raised.

This is how OS/2 dynamically grows stacks. It is also a useful way for application developers to control memory within the application. Since exception handlers can be registered by the application to override the system default handlers, you can handle guard faults as you wish.

Let's look at how you can use these facts and constructs in your programs.

OPTIMIZING DATA IN MEMORY

It is generally a good idea to allocate the memory you will need in your application with your own APIs. By sprinkling `DosAllocMem` and `DosFreeMem` calls all over your pro-

gram, you lose two very important functions. First, you lose the control and placement of data within the pages. Second, you likely will waste memory, since every memory allocation is at least one 4Kb page.

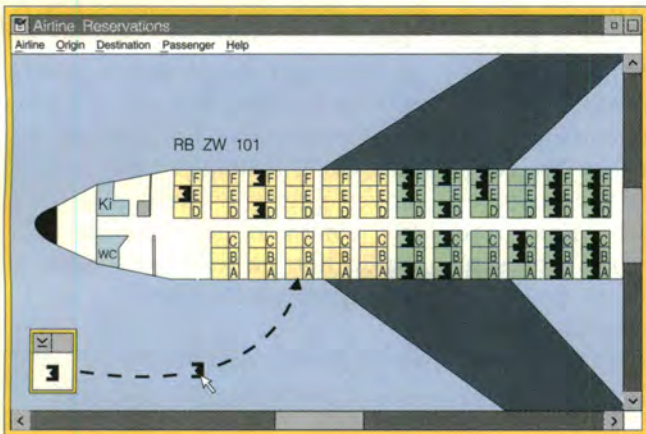
By using your own APIs for application memory allocation and deallocation, you can make intelligent assumptions for a more efficient program design. For example, your memory requests can use special flags—which are designed by you—to group frequently used logical blocks or data structures into a contiguous group of pages. In this way, you can maximize the hit rate of your most-often referenced data with a minimum number of pages in physical memory.

The operating system cannot make these kinds of assumptions about your program. Without good memory control, thrashing can occur, even within the application.

You can also optimize page utilization by keeping an internal memory map and allocating blocks inside pages on a first fit or best fit basis. For example, when a call to your own memory allocation API, for example, `MyAllocMem`, is made in the application, it knows that 200 bytes are needed. Since there are several pages with 200 or more bytes available, a new page is not needed, and a suballocation of an existing page is sufficient. Then you can decide which bit algorithm and which page is best.

This program contains scenes of a graphic nature.

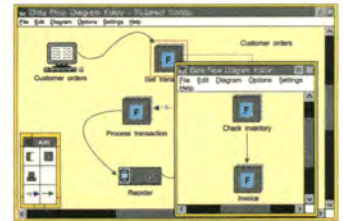
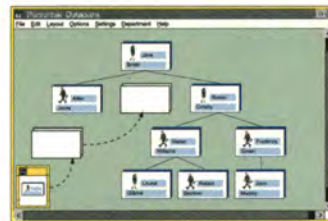
Introducing
Graphics
Interface
Kit/2



But don't worry. It's all in good clean fun. Because now there's a fast, cost-effective way to create graphic, easy-to-use interfaces for OS/2® 2.0 applications and tools. Introducing Graphics Interface Kit/2 (GIK/2), an AD/Cycle® product from IBM Programming Systems.

GIK/2 makes it easy to produce and manipulate symbolic representations of data without writing a single line of Presentation Manager® code. It automatically generates C code, where you can link the graphic symbols to the application data they represent. This not only speeds up your development cycle, it also helps you create applications that are intuitive and easy-to-use. Simplify transaction systems. Make databases easy to access. Bring organizational charts to life. Track inventory with simple icons. Any application can be made easier with the right graphic front-end. And GIK/2 can help you make it happen.

For orders or additional information, please call 1 800 IBM-CALL and ask for Department S71. If you need graphic proof, ask for our evaluation demo which, by the way, is rated G.





You may ask why you need to manage your own memory instead of letting OS/2 do it for you. The operating system will manage memory efficiently for the system and provide you with what you need, but that's all. For example, let's say you have a 2K hash table and a 1K array that are often referenced together. OS/2 doesn't know enough to place them in the same page. But you can do this, because you know the characteristics of your application.

Many of the operating system-provided services can be used within your own memory management scheme. A perfect example is guard pages. Let's say you are writing a program that could really use a large array to handle an incredibly efficient data structure. But to implement this properly, you need a 5MB array in which you'll only hit spots all over, at random. You could register an exception handler to handle guard page exceptions and allocate the entire 5MB object with the guard page attribute set. Your exception handler simply would commit the guard pages as they are hit. This way, unused pages remain uncommitted, yet you have the use of the entire range of addresses in a 5MB linear object.

You can also take the guard page concept a step further. In this exception handler, you can log every time a new page is hit. This allows you to gauge the efficiency of your memory usage and placement of data structures within objects. You can then make algorithmic changes based on the results.

You should allocate a large percentage of the memory you think you will need for the application up front. At first this may seem contrary to the whole concept of efficient memory management. However, if you allocate memory as uncommitted and then suballocate or commit it as needed, you can make the most of what the system provides. Once you have a set of addresses within a set of 4K pages, you can keep the most-often used data in the same physical pages. It is unlikely that these will get paged out, even in memory-constrained systems, since they are being referenced often.

OPTIMIZING CODE IN MEMORY

Now that you've seen how to optimize data within memory pages, what can you do about application code? Code is placed into memory by the OS/2 program loader, so there is not much you can do about code pages' attributes or when they are committed. However, there is something you can do to maximize how much of your code is present in physical memory even in memory-constrained situations.

When organizing code into functions, observe how they interact. Look for the functions that reference each other most often and group them into a small set of pages. You can do this by keeping these functions in the same .C (and hence, .OBJ) files and by carefully ordering statements in the .DEF file when building the .DLL or .EXE files.

By grouping cross-referenced functions together into pages, you can maximize the probability that they will be in memory when needed. When one function is hit and a referenced function is in the same page, both will be resident at the same time. When functions are randomly scattered throughout the code, excess pages may load, hurting your application's performance, as well as the system's performance.

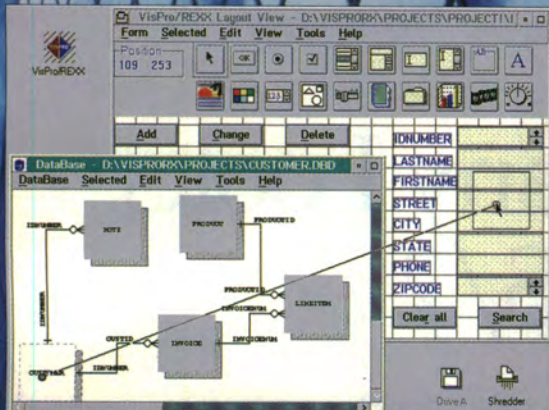
This is the same principle we used for data, but although code pages cannot be as precisely controlled (especially what is placed in them), you can still maximize the hit rate of resident functions.

As you write applications, 32-bit memory management challenges you with new ways of thinking about memory. For the first time, you have the flexibility to use huge data structures, free of contorting segments, expensive pointer arithmetic, and segment register loading. You can optimize performance by understanding how OS/2 manages memory and moves pages, while committing only what is needed to make your application perform well, even in heavily-stressed systems.

David Reich has been with the IBM OS/2 development team since 1987. He has worked on many parts of the system, supported customers and application developers, and traveled the world giving seminars and teaching OS/2. He has recently written *Designing OS/2 Applications*, published by John Wiley and Sons. He can be reached on CompuServe at 76711,632 or via the Internet at speedracer@vnet.ibm.com.

For OS/2 2.x

Get Serious OS/2 Development Power With VisPro/REXX™



Drag and Drop to Build Your Program

VisPro/REXX is a powerful OS/2 tool that is amazingly easy to use.

VisPro/REXX is an easy-to-use visual programming environment that gives you the power to create your own OS/2 2.x GUI applications. Fast.

Whether you want to build a time-saving utility program or a client/server application, VisPro/REXX is the right tool for the job. VisPro/REXX is packed with features that allow you to begin building powerful OS/2 GUI applications almost instantly.

VisPro/REXX Bronze:

- Easiest-to-use visual programming environment
- Only tool with Workplace Shell-enabled drag and drop programming
- More CUA '91 objects than similar programs
- Open environment with SOM Toolkit for building your own objects
- Animated graphical debugger for easier debugging
- OS/2-style help and information line tips
- Access to Database Manager, DB2/2 and Q+E Database Library
- Support for multi-threaded applications

VisPro/REXX Gold includes all of the powerful features of Bronze PLUS:

- Visual database designer for creating DB2/2 applications or reverse-engineering DB2/2 databases
- The *most* CUA '91 objects of any tool in its class
- 3-D business graphics for displaying application data
- MPPM/2 video window for full-motion multimedia
- Ability to add Workplace Shell drag and drop and pop-up menus to your programs
- LAN version available for even more savings

(919) 380-0616

Call us today
with your order
and receive a
free REXX booklet!

Booklet is an excerpt from the
popular 'OS/2 2.1 Unleashed'



\$99 Bronze Edition \$299 Gold Edition

System Requirements:
OS/2 2.x, 5Mb memory and
1.5 Mb hard disk space.

HockWare Incorporated
P.O. Box 336, Cary, NC 27512-0336 (USA)
Phone: (919) 380-0616 FAX: (919) 380-0757

HockWare™

Putting You in Control
Circle Reader Service Number 5

HockWare and VisPro/REXX are trademarks of HockWare Incorporated.
All other company, product and brand names are trademarks and/or registered trademarks of their respective holders and are mentioned for reference purposes only. © 1994 HockWare Incorporated. All rights reserved.



Corporate Study

The insurance industry is rather conservative in its decisions. This cautious attitude extends to business decisions involving data processing. It seems especially significant, then, that OS/2 has become so prevalent in that industry. Provident Life and Accident Insurance, in Chattanooga, Tenn., moved many of its operations from mainframe-based COBOL applications to Workplace Shell and LAN-based OS/2 systems. Here, we talk to members of the Provident team and explore this momentous migration. By BRIAN PROFFIT

Insurance Company Looks to Future With OS/2



Brian Proffit

As late as 1988, virtually all of Provident Life and Accident Insurance's mission critical business applications resided on the mainframe. Management installed a number of PCs but used them primarily for personal productivity applications. At that point, the Information Resources department spent a lot of time thinking about where it needed to be in the 1990s to remain competitive. Bill Quintrell, manager of Microcomputers and Office Systems, knew that some applications were well suited for the mainframe, but many were better suited for and less expensive running on the PC. Reading the tea leaves, Quintrell saw groupware and group computing as signs of changes to come.

"We knew that companies would figure out how to take a single application and move it across the mainframe to PCs—letting the mainframe run its component and the PC component run on the network," Quintrell recalls. "These companies would be the most flexible, have the most choices, and face the least cost. We were in transition from the previous focus of personal productivity to the focus of business systems. Like the revolution that hit in the 1980s on what you could do with PCs, we felt that same productivity boost would hit the corporation."

CHARACTERISTICS FOR SUCCESS

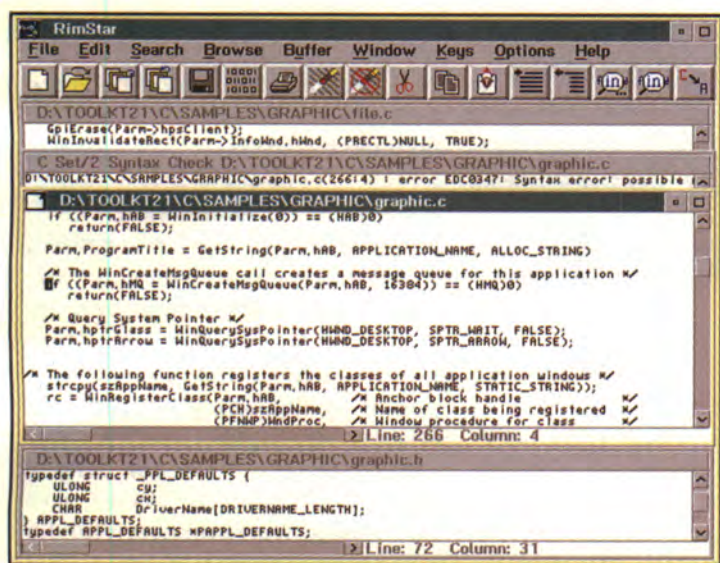
Quintrell had specific, stringent requirements for the new systems. "We value high-quality production business systems. Reliability is important and managing that application is important. It was not practical to throw away our existing investment of applications. We needed fairly robust management business systems, and we needed to support our legacy applications.

"Our staff could not keep up with the explosion of PCs. We could only continue to install and support these PCs if we could automatically install and maintain them. We also needed to install from the LAN. Communications is very important. Multitasking is important and not just for the user's needs. If I'm doing remote maintenance on your system, you need to be able to keep working. The users knew GUIs were becoming important, but on a DOS base it was complex, with device drivers and memory managers. They had to learn too much about the PC itself, rather than how to use it as a tool.

"It was an easy decision. We felt that OS/2 was the only choice for our environment. It was a more mature operating system and allowed us to buy or write robust business applications. Most importantly, there were absolute requirements to create

Brief's performance is over. Meet the new star.

The RimStar™ Programmer's Editor V2.0



If you're looking for a fully configurable, professional OS/2 PM programmer's editor to replace your current text-mode editor – we have what you have been looking for!

No hassle changing editors

We know changing editors can be a major hassle. You don't want to relearn a new set of keystrokes no matter how good the product. Well, you don't have to – we have Brief, Epsilon, Multi-Edit, SlickEdit, PWB and Borland IDE keyboard mapping waiting for you. If you're not using one of these, let us know and we will create the mappings you need.

Features designed to increase productivity

It has all the features you have come to expect, plus special features designed to anticipate your needs and increase your productivity. Features like a 'C' source browser that eliminates those time consuming searches for functions, global variables, typedefs, and macros by providing you with instant positioning to both definitions and references. Using our powerful 'C' macro language you can customize your programming environment to suit your individual needs.

All the features you need

- ✓ Complete 'C' macro language
- ✓ Auto-indent & template editing
- ✓ No limits on files, file size, or windows
- ✓ Bookmarks
- ✓ Line lengths to 16K
- ✓ Timed auto save
- ✓ Access PDK help for function under cursor
- ✓ Multi-threaded for no waiting
- ✓ Configurable tool bar
- ✓ Import/export to system clipboard
- ✓ Keystroke record/playback
- ✓ Block indent/outdent
- ✓ Brace matching
- ✓ Customizable menus
- ✓ Column, line and block selection, search and replace
- ✓ Integrates with Workframe/2
- ✓ Source browser for 'C'
- ✓ Unlimited undo and redo
- ✓ Multi-buffer regular expression search and replace
- ✓ Compile and jump to errors
- ✓ Complete on-line help
- ✓ Save and restore state between sessions
- ✓ OS/2 2.x 32 bit PM Multi-document interface

"My copy of Brief has been permanently retired. Keep up the good work!" - A.L.

Price \$299.00

Plus Shipping & Handling.

To order call (603) 778-2500.

90 day money-back guarantee.

Circle Reader Service Number 6

RimStar Technology, Inc.

91 Halls Mill Road
Newfields, NH 03856-0938

All products and company names are trademarks or registered trademarks of their respective holders.

© 1993 RimStar Technology Inc.

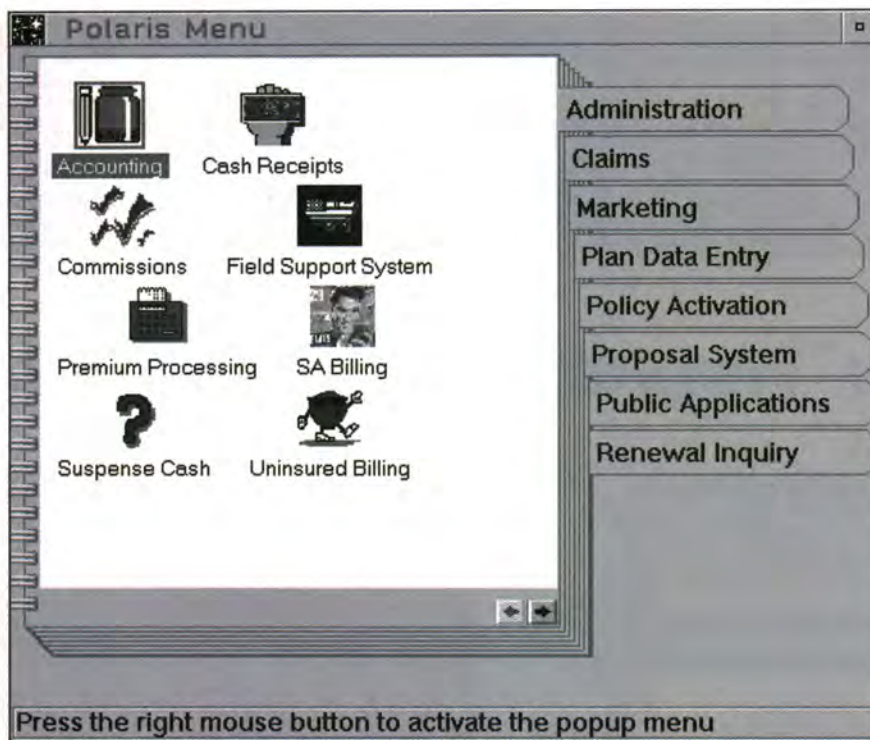


Figure 1. A view of Polaris

large-scale, line-of-business applications that were founded in cooperative processing. If I am to split an application between the mainframe and the micro and deliver it faster or lower the cost, I need a PC system with strong communications, better data management, and higher reliability than the users themselves require. Clearly, corporate developers would absolutely demand and data processing professionals would absolutely require something like OS/2."

EVOLUTION, NOT REVOLUTION

To get where Provident wanted to be was a challenge. Like most companies, Provident couldn't afford to throw out all of its existing PCs and buy new ones. Instead, it threw out the underlying philosophy that had guided its PC purchases.

"We drove a stake into the ground in 1988," Quintrell recalls. "From then on, everything we invested in—hardware, software, and people—would fit in with the enterprise-connected, host-inte-

grated OS/2 environment we wanted. We didn't attempt to convert our existing personal productivity systems to the new model—it wasn't practical. So, we created parallels. On one side was the existing base of DOS programs that immediately went into maintenance mode—no enhancements or updates. On the opposite side was the new path of buying, building, and installing OS/2 and the networks we needed to stay competitive."

The commitment was made near the end of 1988. In early 1989, Provident began to make the changes. "The first piece we installed was LAN Server. It was November, 1989 before we started installing OS/2 user stations. Our core products were not available on OS/2—it wasn't until then that OS/2 was ready for what we needed.

"From that point, we could architect our base components. The changes meant that all servers go into the glass house and all LANs are connected by a corporate backbone. That's a lot of wire. The operations

staff was trained for backup and disaster recovery. There were new availability needs for our servers. As you can see, there was a lot of infrastructure that had to be created."

Some of the cost-conscious users questioned the data processing staff about the size and power of the new PCs. They didn't see the need for that much power for the personal productivity applications they were running.

"That was purely an issue of education," laughs Quintrell. "We communicated that we knew what we were installing wasn't being exploited then but would be in the near future." Slowly, each PC was upgraded to an OS/2 LAN system.

FORSAKING THE MAINFRAME

In early 1990, Provident reorganized, splitting Long Term Disability business from Group Medical and creating a separate operating unit. Then, the disability business was supported by the Group's mainframe systems, and the new operating unit wasn't satisfied. Jim Cook, who is director of Disability Systems, says that this restructuring provided a perfect opportunity to review the data processing. There were also concerns about having Provident's data processing needs managed by a different business unit. Platforms other than OS/2, however, were never a consideration.

"The decision was whether we should do our data processing on the mainframe or the PC," Cook says. "We looked at what it was costing us mainframe-wise, what it would cost on the PC, and the type of business we were doing. We looked at our needs and things like graphical user interfaces that make it easier and more intuitive for the user. Based on those considerations, we decided that we should develop in the PC environment. We felt we could build systems faster and that they would be more user-friendly. We also felt we could

be more flexible and responsive to the changing business conditions more quickly in the PC environment. After that, OS/2 was the only choice. Bill Quintrell's group had OS/2 networks set up, and they recommended that OS/2 was the only operating system that would really be able to handle our applications."

EARLY TOOLS CREATED PROBLEMS

Following the Information Resources group's lead, Cook's organization began its development using Borland's Paradox on OS/2 1.1. The group developed its proposal generation and contract issuance systems but concluded, in the third quarter of 1990, that Paradox wouldn't be able to handle the volumes they needed. It also could not provide as

much function as was required. After searching, they switched to IBM's Database Manager/2.

"We started over again, in terms of our application development," Cook recalls. "We looked at building an integrated database and data dictionary and started from the bottom. Over the next year and a half we developed most of the processing systems that we needed to get off the mainframe."

Cook's organization deployed the software through a gradual rollout, like Quintrell's hardware transition "We did it piecemeal, with the proposal generation and contract generation first. We followed up with the administrative systems: the claims processing system, financial systems, and accounting systems."

Unfortunately, the switch to

Database Manager/2 may have fixed the underlying data storage problems, but the actual application development was still a problem. All the programming expertise was in mainframe COBOL, and the learning curve and tool immaturity worked against the developers.

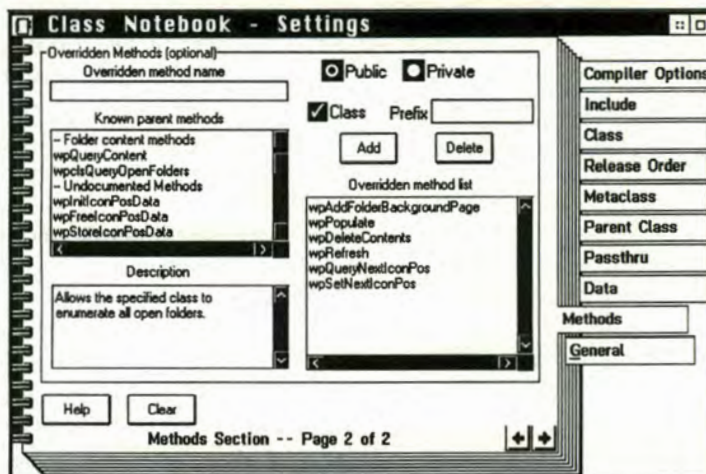
"We encountered a lot of problems because we developed this with Micro Focus COBOL," Cook says. "At that time, the API calls were not converted to COBOL; they were all in C. We couldn't find documentation on how to interface to the APIs with COBOL. We had to work our way through virtually all of those APIs and figure out how to convert them to COBOL. The technical support from Micro Focus has gotten better, but at the time we were on our own to figure out how the inter-



SOM Tool!

The Object Factory

- The fastest and easiest way to develop SOM
- Single click SOM compilations
- Double click overrides of all inherited methods
- Drag drop class associations
- Full on-line documentation for SOM classes and methods
- Includes undocumented SOM classes and methods
- Implemented totally via SOM



The Object Factory is the full featured SOM development environment you have been waiting for. Tapping into the unlimited power inside OS/2 is now within reach. The Object Factory gives you the development advantage you require.

System Requirements: OS/2 2.x and SOM 1 or SOM 2.

5085 Forest View Trail Douglasville Georgia 30135 (404)942-8699

SYNAPTEC, INC.

Circle Reader Service Number 7



nals of the code worked. The concept of messaging and the kind of messages generated were foreign to us. There were months where we were saying 'we can't do it,' but we kept moving forward. One by one, we solved the problems."

Provident did get help with the graphical interface code, though.

"We used CASE:PM for COBOL [since renamed KASE:PM], which helped quite a bit. That's basically a code generator that created the code for the graphical portion of the program. We were the first users of the product to develop production systems with it. We probably would have given up without it."

Interestingly, when they decided to do programs in C, they selected GPF Systems' Gui Programming Facility (Gpf) as their code generator, rather than KASE:PM for C.

"One factor was cost," Cook says. "Gpf was cheaper than KASE:PM. And the people that evaluated and developed our standards liked Gpf better. They felt it was better on the regeneration than KASE:PM was. Since then, Kaseworks released a VIP version that is probably as good as Gpf, but we don't see any reason to change, and it is still more expensive." They also got help from Intersolv's PanLCM version control system, essentially a PC-based version of their Panvalet mainframe product.

UPDATING THE INFRASTRUCTURE

With the assistance of Information Resources, Provident built the hardware base required simultaneously. It has 125 users connected to the network, virtually all of which are PS/2 model 70s. Provident has six PS/2 model 95 servers on an IBM token ring network, one of which is a domain controller, one a print server, and four database servers. Some of the users—particularly actuaries who use large spreadsheet models and Pioneer Software's Q+E

query tool extensively—need more power and are updating their systems with new 486 motherboards.

Provident updated the software base as well. They migrated to OS/2 1.3 in January of 1992, moved to OS/2 2.0 later that year (made easier by their early commitment to hardware that supported that platform), and updated to OS/2 2.1 last year. The move to the 32-bit platform also allowed Provident to upgrade their database to DB2/2.

GETTING USERS UP TO SPEED

Due to a corporate commitment to OS/2, Quintrell's organization provided training on the operating system at its corporate training facility. This—combined with aggressive use of Workplace Shell features such as notebooks and the Information Presentation Facility for on-line help—made Cook's job of training users on the new Long Term Disability system, called Polaris, simpler. Polaris is shown in Figure 1.

"The training needs were minimal because the systems were so intuitive in their design," Cook says. "Generally, we could just bring the user group into our large-screen conference room and walk them through the system and answer their questions. Then they went back to their desks and worked with the system in test mode, and in most instances that alone was sufficient for training."

This is a nice contrast to the efforts required for mainframe systems. "The training needs have been reduced. The users take to it quickly. They find it fun to be working on these systems. With any new toy there is greater receptivity, but they have been very receptive to it."

To make sure the excitement doesn't fade after the newness wears off, Cook surveys users quarterly. "The feedback is very positive," Cook reports. "They say they could never go back to the mainframe, they love how easy it is to

use, how much information they have available, and so on."

TAKING IT TO REMOTE OFFICES

Now that Polaris has become popular at headquarters, Cook is challenged to get it to remote offices. In fact, Cook says this is "the biggest issue we face in 1994. We have not yet given remote access to our network, and because of cost considerations and the way our field offices are set up, it's not cost effective to do that with the technology that's available right now."

"We looked at the bridge and router technology. We estimate it will cost anywhere between \$12,000 and \$20,000 per office to do that, and in a lot of our offices we may only have one machine that needs connectivity to the Long Term Disability network. That's not very cost-effective. We could do it on a dial-up basis, but we have reservations about that also: from a security standpoint, we're concerned about exposure. Even if it's only one workstation, if it's in California and has to be connected eight hours a day, it is expensive as well."

"We tested IBM's LAN Distance as a potential dial-up solution but had problems with it. We also looked into using Distributed Data Connections Services/2 (DDCS/2) communications software, but it didn't work well with our applications. We developed [the applications] to exploit OS/2 with multiple programs running concurrently. In the way DDCS/2 works with synchronous data link control (SDLC), though, it recognizes that as multiple sessions. However, SDLC will only recognize a single session. We hoped to shadow our databases on DB2 and let the remote offices come in through their mainframe connection. However, [SDLC] couldn't recognize multiple sessions, so it would disconnect. We may still do that, but we would have to redesign our applications."

OS/2 PROVIDES FOR PROVIDENT

Quintrell identifies four separate needs in an information system. "We need to see that the user's needs are met, the line level or department needs are met, the corporation's heads of individual business units' needs are met. We also need to meet the data processing audit requirements. The best approach is to deliver a system that balances the four needs. You can't sacrifice any of them. Other operating system and application vendors don't understand the need for this balance as well as IBM does; they focus only on the user. The corporate market clearly has a different objective than the user. When you buy a PC and put it on the desk, the hardware is the cheapest part. The user administration is by far the greatest.

"OS/2 is a balanced system, with DOS, Windows, 32-bit, and the strongest communications of any operating system. It has high reliability and security. If you're just going for personal productivity, you may look at DOS/Windows. If you want something with high-end programming, you might get an engineering station. If you want high reliability and processing power, you might look at mainframes. If you have to balance all of these needs, OS/2 is the clear choice."

Still, while happy with OS/2, Quintrell hopes that IBM is not complacent. "We need for IBM to keep OS/2 focused. Embedding technology is critical. They've got a lot of talk and presentations, but to make people wait until 1995 for something similar to what was in version 1 of OLE is inexcusable. IBM is working on bidirectional printing and more printer drivers than the 300 already there, but what are they doing to improve the product for those of us printing already? I'd like to see my word processor put out page one on letterhead and page two on plain paper. How can OS/2 prevent an application from a mid-

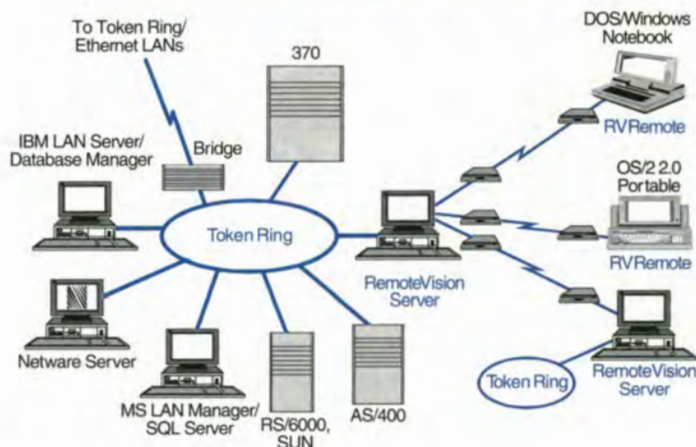
job forms change? Wordstar was doing that when Moses was printing the commandments. The old IBM would have been responsive to that. The new IBM is concentrating too much on the mass market."

Brian Proffit has been in the computing industry for 20 years. He was part of IBM's

OS/2 team but left to become the director of PC Week's Corporate Labs. He is now president of Visionary Research. Proffit is the author of OS/2 Application Development Tools and OS/2 Inside & Out. He has written for PC Week, Dr. Dobb's Journal, and Programmer's Paradise and is a contributing editor for OS/2 Magazine. He can be reached on CompuServe at 75300,1466.



RemoteVision™ MultiPort™ Servers. The Proven Solution For Remote Token Ring Networking.



- Extend LANs transparently over asynchronous modems to remote workstations and workgroups.
- Run DOS, Windows™, & OS/2™ apps written to multiple protocol stacks on remote nodes.
- Operate remote computers as full-function LAN nodes.
- Connect to remote offices with LAN-to-LAN dial-up networking.
- Add remote connectivity services using existing LAN machines.
- Access remote PC servers and hosts without additional host gateways.
- Up to 64 ports on a non-dedicated server

RemoteVision makes it easy and affordable for DOS, Windows and OS/2 users in remote locations to connect to Token Ring or other LANs using dial-up modems. Response time remains fast, applications don't need changing, and users won't need retraining. So you can turn your remote machines into full-function workstations. With RemoteVision. To immediately find out more, or to take advantage of our "Try Now, Pay Later" 30-day money-back guaranteed trial offer, call today.



1265 Montecito Ave., Ste. 101, Mountain View, CA 94043
Tel: (415) 965-8607, Fax info: (415) 965-8607, (then press 2)
CompuServe 71612, 1253. Trademarks are from their respective companies. © 1994 Token Technology, Inc. 001-1

Circle Reader Service Number 9



The NFS product on OS/2 does not support remote file locking. As a result, mutual exclusion is not enforced among the different components of a distributed system when they attempt to update the same file via NFS. This article presents some practical algorithms to cope with the problem. The algorithms represent a variety of performance vs. convenience and safety tradeoffs one can make. By HARI MADDURI

Locking Network File System Files on OS/2



Hari Madduri

The Network File System (NFS) is a distributed file system that allows users to access remote files as if they were local files. Using NFS, a server machine exports a directory tree of files, and the client machines access those files via the mount command.

The NFS product on OS/2 does not support remote file locking. As a result, mutual exclusion is not enforced between a local process accessing a local file and a remote process accessing the same file via NFS. Lack of mutual exclusion can cause problems for distributed applications that need to update shared files in exclusive mode. In this article, we present very practical algorithms to cope with this problem. Many of these algorithms do the job, but they cannot be proven correct. That is, theoretically, there are situations under which they fail, while in practice they might work all the time.

The algorithms presented in this article are of special value to users of persistent SOM and DSOM objects whose data is stored in NFS files, since it is the user's responsibility to protect such data from concurrent updates by multiple processes or workstations.

THE PROBLEM

Assume that we have a shared file (in general, a network-wide resource) that is

used by multiple processes in a distributed application. The processes may be located on multiple machines. Processes located on remote machines access the shared file through NFS. The file's local machine exports the directory (or drive) containing the file, and the remote machines "mount" the exported file system onto a node in their local directory tree. Assume further that no process uses the shared file for more than Delta seconds at a stretch, where Delta can be set depending upon the application. All processes are required to adopt the following protocol to use the shared file:

1. Open the shared file SF with exclusive write lock.
2. Use the file (a maximum time limit of Delta seconds).
3. Close SF and release the exclusive write lock.

The problem is that when a remote process and a local process both try to use the file, they both can get a lock to the file and start updating the file. (This is a problem with the current implementation of TCP/IP and NFS product [v.1.2.1]. It may not be a problem in the future releases.)

THE SOLUTION

We will build a comprehensive solution by starting with a simple algorithm and enhancing it step by step. Depending on



the requirements of the application domain, any intermediate point in this gradual refinement can serve as a satisfactory solution. For ease of explanation, we present the initial algorithm and all refinements using either pseudocode or simple flow charts. Finally, we present an implementation of a more complete solution in C.

ALGORITHM I

The first solution is based on the simple idea that files in a network file system can be used as network-wide semaphores. In step one of the protocol, we can create an auxiliary file, Lock file, and use its presence to indicate that the shared file is being used. The Lock file can be deleted in step three. If two processes, one local and the other remote, try to create the Lock file at the same time, only one will succeed and the other will fail. This is so because file creation is supposed to be atomic in all file systems. The failed process has to wait at least Delta seconds (assuming Delta is the maximum time for which the shared file can be used by a process) and retry. Thus, the algorithm is:

1. Create a Lock file L1; if not successful, **return(failure)**.
2. Open the shared file SF.
3. Use the file (a maximum time limit of Delta seconds).
4. Close SF.
5. Delete L1.

When the algorithm returns with a failure indication, we wait for Delta seconds and reexecute the algorithm. This algorithm is simple and works so long as all processes follow the protocol strictly. Also, if we are not concerned about tolerating process and machine failures, then this algorithm is sufficient for coping with the locking problem. However, if we want to be fault tolerant, then we run into problems. Consider what happens if a process creates L1, uses SF, and dies before deleting L1. The remote processes wanting to create

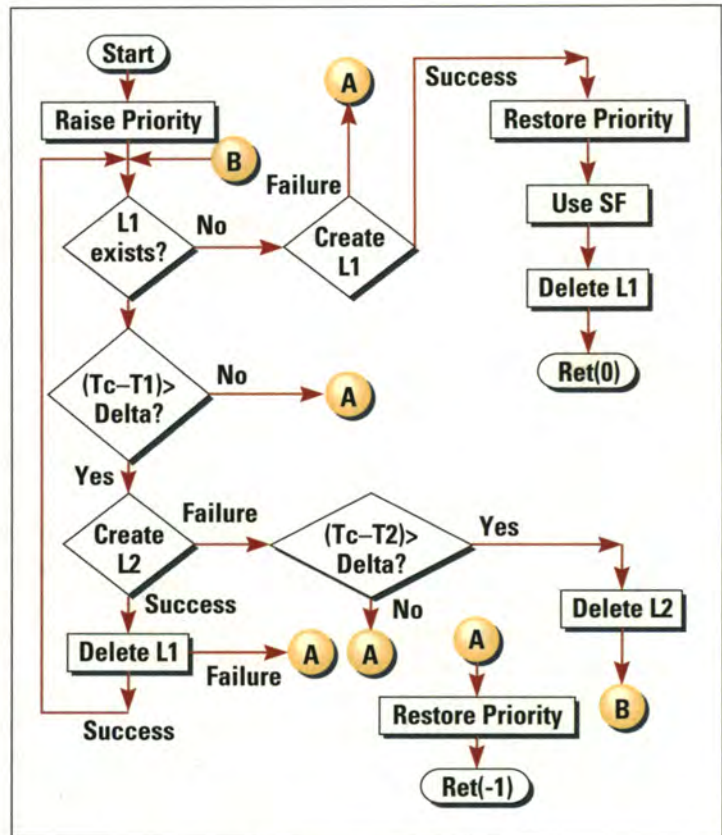


Figure 1. An illustration of Algorithm III

L1 will wait forever since they keep failing to create. You can modify the algorithm so that it repeats only a finite number of times and then quits with some error message to the user. The user can then manually delete L1 and restart the process. It would be nice to be able to do this automatically without user intervention. The next algorithm attempts this automation.

ALGORITHM II

This algorithm builds on the previous one. The trick here is to recognize an old Lock file and delete it.

1. Create a Lock file L1.
2. Open the shared file SF.
3. If not successful, do LockRemoval-Algorithm and **return(failure)**.
4. Use the file (a maximum time limit of Delta seconds).



5. Close SF.
6. Delete L1.

LockRemovalAlgorithm I:

1. T1 = Creation time of L1; TC = Current time
2. if ((TC - T1) > Delta) L1Old = True; else L1Old = False;
3. if L1Old, Delete L1;
4. return;

As before, when it returns with failure, we wait for Delta seconds and reexecute the algorithm. This algorithm looks reasonable, but it has two problems. One is that if the clocks on different machines are not synchronized (which is not uncommon), then it is possible to have one of the processes delete L1 while another process is still using the shared file SF (that is, erroneous deletion of Lock file L1).

The other problem is that even if the clocks are synchronized, an erroneous deletion of L1 can occur. For example, assume processes P1 and P2 (both remote) are executing LockRemovalAlgorithm at the same time. Further assume that both of them have concluded that L1 is old and are about to delete L1. Let's say that P1 gets swapped out at this point and P2 continues. P2 deletes L1, goes back, creates L1, and starts using SF. P1 now starts running again and deletes L1. But this L1 is new, since it has been created by P2!

The first problem of unsynchronized clocks on different machines can be solved. Instead of calculating Delta as simply the difference between the current time and the time of creation of the lock file, you can use the following algorithm. Assume that there is always a timestamp written in the lock file.

1. Open the lock file and write the current timestamp into it.
2. Wait for Delta seconds and see if the timestamp in the lock file has changed.
3. If it hasn't, then conclude that

```
/*                                pprlock.c                                */
/* This module implements a network-wide lock for NFS files on
OS/2. To use this lock module, see the test program tstpprlock.c */

int process_lock(char *lckfilstem, int *result){
int file_status;
LOCKSTAMP lockstamp1;
LOCKSTAMP lockstamp2;
int rc1;
char lckfile[FILENAME_MAX];

lckfile[0] = '\0';
strcat(lckfile,lckfilstem);
strcat(lckfile,".lk1");
if ((rc1 = process_lock1(lckfile, &lockstamp1, &file_status)) != 0) {
    *result = LOCKF_FILE_LOCKED;
    return(0); }

if ( file_status == LOCKF_FILE_AVAILABLE ) {
    *result = LOCKF_FILE_AVAILABLE;
    return(0); }
/* GOOD RETURN CODE BUT FILE UNAVAILABLE */
*result = LOCKF_FILE_LOCKED;
DosSleep(Usage_time_max); /* sleep until the current user finishes. the
Usage_time_max_ represents the maximum sleep time of Delta seconds */

set_priority_high();
if(process_lock1(lckfile, &lockstamp1,&file_status) != 0 ) {
    restore_priority();
    return(0);}
/* GOOD RETURN CODE ON SECOND TRY */
if ( file_status != LOCKF_FILE_OLD ) {
    *result = file_status;
    restore_priority();
    return 0; }
/* L1 is outdated */
*result = LOCKF_FILE_LOCKED;

if (process_lock2(lckfilstem, &lockstamp2, &file_status) != 0) {
    restore_priority();
    return(0);}

if ( file_status != LOCKF_FILE_NEWL2) {
    *result = file_status;
    restore_priority();
    return 0; }
/* An L2 exists */
*result = LOCKF_FILE_LOCKED;
DosSleep(Usage_time_max);

if (process_lock2(lckfilstem, &lockstamp2, &file_status) != 0){
```

Figure 2. An implementation of the algorithm in C (continued on page 22)

Introducing IBM C Set ++ V2.1 and KASE:Set.

C Set ++ V2.1
including Kaseworks KASE:Set.
Starting at \$393 (U.S.) - CD-ROM Version.

Object oriented development.



C Set ++ Version 2.1 for OS/2® from IBM Software Solutions is one of the most complete object-oriented application development packages you can buy.

Plus.

Its 32 bit C/C++ compiler, with its advanced code optimizer *and*

WorkFrame/2 V2.1 - a

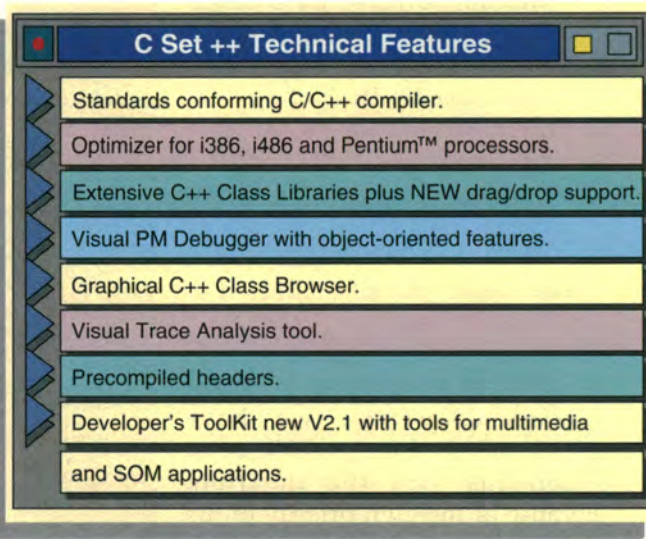


C Set ++

completely new development environment based on OS/2's object-oriented Workplace Shell - allows you to create up-to-the-minute mission critical applications.

You also receive a specially tailored copy of Kaseworks KASE:Set, a visual design and code generation tool which allows you to build visual applications even more quickly, with a minimal investment in learning time.

Plus!



To order C Set ++ for OS/2 including KASE:Set, or for further information call 1-800-342-6672 (U.S.A.) or 1-800-465-7999, ext. 676 (Canada). Or contact your local IBM software dealer.



the lock file is old. If it has, then it is not old.

(For added robustness, include the process ID of the attempting process in the timestamp.)

The second problem of erroneous deletion of L1 is hard to solve. In a distributed system, without the aid of any locking or reference entity (such as a process or an agent), it is impossible to be sure that the two operations of checking the file's age and deleting the file are done on the same file! The fundamental problem is that you don't have an atomic action that brackets the two operations (determining L1's age and deleting L1).

We can, however, try to minimize the probability of an erroneous deletion. Consider this algorithm, LockRemovalAlgorithm II:

1. T1 = Creation time of L1; TC = Current time
2. Set process Priority to its maximum limit
3. if ((TC - T1) > Delta) L1Old = True; else L1Old = False;
4. if L1Old, Delete L1;
5. Restore Priority, return;

With this modification in the LockRemovalAlgorithm, we have greatly reduced the probability of erroneous lock file deletion. Because of the high priority of the process the steps three and four are executed in quick succession, thereby minimizing the chance of process scheduling playing havoc at this point. It is still possible for two or more processes to execute the preceding algorithm and arrive at step three at the same time. They may all conclude that L1 is old, but only one of them successfully deletes L1 and the others fail on delete.

They can still return to the main algorithm and retry creating L1. We say the probability is greatly reduced since processes cannot decide that L1 is old and wait for too long before actually deleting L1 (due to their high priority, they

```
restore_priority();
return(0);}
*result = file_status;
restore_priority();
return(0);
}

void process_unlock(char *lckfilstem){
int rc;
char lckfile [FILENAME_MAX];

lckfile[0] = '\0';
strcat(lckfile,lckfilstem);
strcat(lckfile,".lk1");
if (rc = DosDelete(lckfile))
    printf("Failed to delete lock file. Mutual exclusion is probably violated\n");
}

int process_lock1(char *lckfile, LOCKSTAMP *l1stamp, int *status){
LOCKSTAMP newl1stamp;
int rc, rc1;
int lckfd;
FILE *lckfp;

*status = LOCKF_FILE_LOCKED;
if ( (lckfd = open(lckfile, O_CREAT|O_EXCL|O_RDWR, S_IWRITE)) < 0 ) {
    switch (errno) {
        case EEXIST:
            if ( (lckfd = open(lckfile, O_RDWR, 0)) < 0 ) return(-1);
            if ( (lckfp = fdopen(lckfd, "r+")) == NULL ) {
                rc1 = close(lckfd);
                return(-1); }
            rc = getlockstamp(lckfp, &newl1stamp);
            rc1 = fclose(lckfp);
            close(lckfd);
            if (rc != 0) return(rc);
            if (lockstamp_compare(l1stamp,&newl1stamp)!= 0) {
                memcpy(l1stamp,&newl1stamp,sizeof(LOCKSTAMP));
                return(rc); }
            else /* L1 is old */ {
                *status = LOCKF_FILE_OLD;
                return(0); }
            break;
        default: return(-1);
    } /* end switch */
} /* end if clause */
else /* L1 was created */ {
    if ( (lckfp = fdopen(lckfd, "r+")) == NULL ) {
        close(lckfd);
        return(-1); }

    rc = putlockstamp(lckfp);
    rc1 = fclose(lckfp);
```

Figure 2. An implementation of the algorithm in C (continued on page 24)

VNR KNOWS OS/2® PROGRAMMING

OS/2® V2 C++ Class Library:

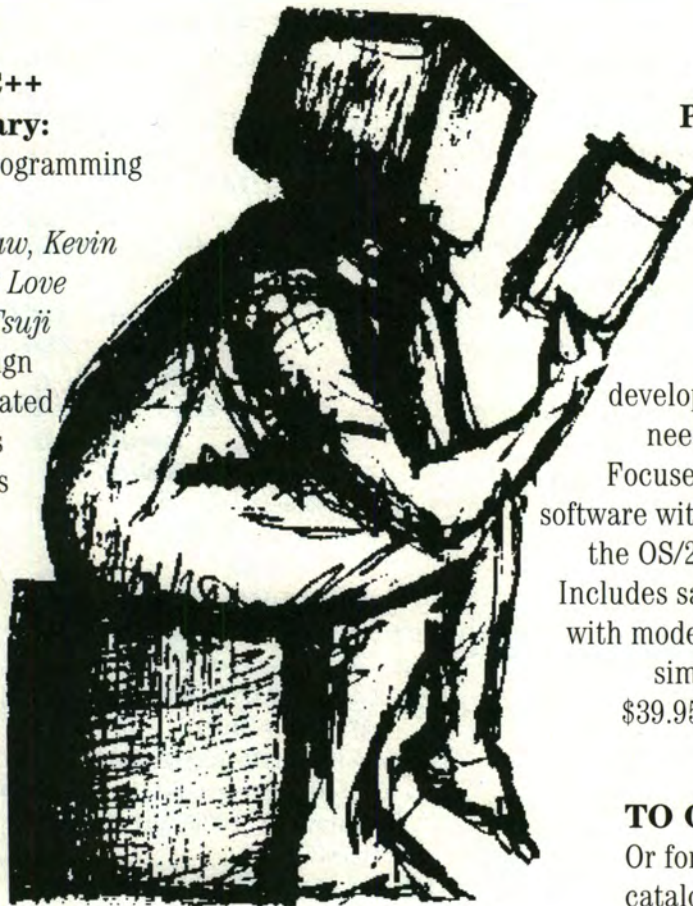
Power GUI Programming
with C Set++

*by William Law, Kevin
Leong, Robert Love
and Hiroshi Tsuji*

From the design
team that created
the IBM Class
Library, this is
the only
reference and
tutorial you'll
ever need on
the subject.

\$36.95

0-442-01795-2



C+ and C++ Programming in the OS/2® Environ- ment

*by V. Mitra
Gopaul*

All the IBM
development tools you
need...in one place.
Focuses on developing
software with C and C++ in
the OS/2® environment.
Includes sample programs
with models ranging from
simple to complex.

\$39.95 0-442-01240-3

TO ORDER:

Or for a complete
catalog of all VNR
OS/2® books, call
1-800-544-0550
(Dept. 1606) or
GO VNR on the
Compuserve®
Network.



Van Nostrand Reinhold

115 Fifth Avenue, New York, New York 10003
73373.64@compuserve.com

OS/2® is a registered trademark of the IBM Corporation.

Circle Reader Service Number 12



must run quickly and finish the steps). However, the algorithm is still not bullet-proof since there can still be communication delays for NFS messages accomplishing the file operations.

ALGORITHM III

We present another algorithm that is designed to further increase safety against erroneously allowing two processes to use SF. We use the idea of a second lock file, L2, to ensure that only one process gets the right to delete the original lock file, L1. Also, in this algorithm we combined the lock removal algorithm with the main algorithm.

Figure 1 illustrates the new algorithm. L1 and L2 are lock files. T1 and T2 are their creation times. Tc is the current time. Delta is the upper bound on SF usage time. A return value of 0 means success and -1 means "wait (at least Delta seconds) and retry."

The improvement here is that when we find L1 to be old, we don't attempt to delete it directly. Instead, we create another lock file, L2 and then delete L1. The creation of L2 ensures that when several processes race to delete L1, one succeeds and the rest fail and retry creating L1. Even if they lose the race to create L2, it is still possible for one of those processes to create L1 successfully and access the shared file SF.

The problem of a having an old lock file in the current directory still persists. We have devised a method for getting rid of a leftover L1, but we face an equivalent problem with a leftover L2. To cope with this problem, we use LockRemovalAlgorithm II on L2.

This algorithm is better than the earlier one for the following reasons. LockRemovalAlgorithm II, like the earlier algorithms, still suffers from a problem: we cannot be completely sure that the file we are deleting is the same as the one

```
close(lckfd);
if ( (rc == 0) && (rc1 == 0) ) *status = LOCKF_FILE_AVAILABLE;
return(rc);
}
return(0);
}

int process_lock2(char *lckfilstem, LOCKSTAMP *l2stamp, int *status){
char lckfile [FILENAME_MAX];
LOCKSTAMP newl2stamp;
int rc;
int lckfd;
FILE *lckfp;
*status = LOCKF_FILE_LOCKED;
< ... Similar to process_lock1 ... >

if ( (lckfd = open(lckfile, O_CREAT|O_EXCL|O_RDWR, S_IWRITE)) < 0 ) {
switch (errno) {
case EEXIST:
< ... SIMILAR TO PROCESS_LOCK1 ... >
if ( lockstamp_compare(l2stamp,&newl2stamp) != 0
) {
memcpy(l2stamp,&newl2stamp,sizeof(LOCKSTAMP));
*status = LOCKF_FILE_NEWL2;
return(rc); }
else /* L2 is old */ {
remove(lckfile);
return(0); }

default:
return(-1);
break;
} /* end switch */
} /* end if clause */
else /* L2 was created */ {
< ... SIMILAR TO PROCESS_LOCK1 .... >
lckfile[0] = '\0';
strcat(lckfile,lckfilstem);
if (remove(strcat(lckfile, ".lk1")) == 0 )
*status = LOCKF_FILE_READY;
return(0);
} /* end of else */
return(0);
}
```

Figure 2. An implementation of the algorithm in C (continued from page 22)

we looked at in the preceding steps. This problem is inherent in the environment. However, we have the following improvement. L1 is protecting the resource (SF) directly and L2 is only protecting the deletion of L1. Since we have

shifted the problem from L1 to L2 now, the probability of erroneous access to the shared resource (SF) is greatly diminished. It is true that multiple processes can now attempt to delete a leftover L2, but regardless of which succeeds in

Zap OS/2 client/server problems with the power of RemoteControl/2.



A centralized support solution that'll energize your OS/2 applications.

With the success of OS/2® comes the burden of managing a rapidly expanding environment of OS/2-based client workstations and database servers. To meet the challenge, Compuware introduces RemoteControl/2™ to help you access, navigate, and control a proliferating OS/2 environment from your desktop.

Database administration and system management in one dynamic tool.

Managing data security, database configurations, backups and recoveries is only half the client/server management battle. What about file distribution? UPM configuration? Job scheduling? System and network configuration? And DASD and memory management? With RemoteControl/2, you get it all.

Bridge the gap between LANs and WANs.

While accessing remote workstations and servers on a LAN is important, you also need to access remote LAN systems through WAN connections. Remote Control/2 provides the connectivity to access systems wherever they are using your network infrastructure.

Make client/server pay off by controlling escalating support costs.

RemoteControl/2 puts an end to frequent support staff travel, eliminates "overnight" diskette mailings, and can even eliminate the need for additional local support at remote sites. With RemoteControl/2 you can control and plan the support of quickly expanding applications no matter how far your enterprise extends.

**Meeting needs, exceeding expectations.
Compuware client/server products.**



COMPUWARE®
Sophisticated Software Used Every Day

*For more information on
OS/2 Client/Server Management,
call 1-800-535-8707 or 313-737-7300*

OS/2 and DB2 are registered trademarks of International Business Machines Corporation. RemoteControl/2 is a trademark of Compuware Corporation.

Circle Reader Service Number 13



deleting L2, we achieve our desired goal of getting rid of L2.

Even this algorithm can be shown to fail by constructing a particular sequence of events. For example, assume that three processes, P1, P2, and P3, decide to delete L2, and the first of them, P1, deletes L2 and reexecutes the algorithm. Let's say P2 fails to delete L2, reenters the algorithm, sees an old L1, and waits just before creating L2. Let's say P1 creates L2, deletes L1, reenters the algorithm, creates L1 again, and starts using the shared file SF. P3 can now delete L2, clearing the way for P2. P2 will now execute the same steps as P1 and will access the shared file SF concurrently with P1!

Of course, this requires the process scheduling to delay processes by the right amount to make it all happen, but it is possible, theoretically. Even if you raise the priority of the processes executing this algorithm to be the highest, you can still be delayed by unknown amounts due to the network. But practically speaking, the probability of this algorithm failing is extremely small. Figure 1 illustrates Algorithm III.

AN IMPLEMENTATION

An implementation of the algorithm in C is given in Figure 2. (To conserve space in the article, we have not shown the implementation of some simple functions and some declarations.) The implementation incorporates the algorithm for taking care of unsynchronized clocks and also minimizes the undesirable interference from process scheduling. The implementation also does some simple optimizations rather than a straight one-to-one mapping of the algorithms presented. The first file, `pprlock.c`, is a module that implements locking without any specific knowledge of the shared resource. It only requires

```
/*                                tstpprlock.c                                */
/* This driver program tests the lock algorithm in pprlock.c
It accesses a shared file "myfilename.ext" and writes the current
value of system time into it. It repeats this action 20 times.
By running two or more copies of this program concurrently we can
create contention for the shared file and see how the algorithm in
pprlock.c solves the problem. */

main ()
{
    int fd, lkfd;
    FILE *sffp;
    int i, index,
    lkresult;
    TIMEVAL tp, oldtp;
    set_usage_time_limit(SF_DURATION); /* set this in lock module */

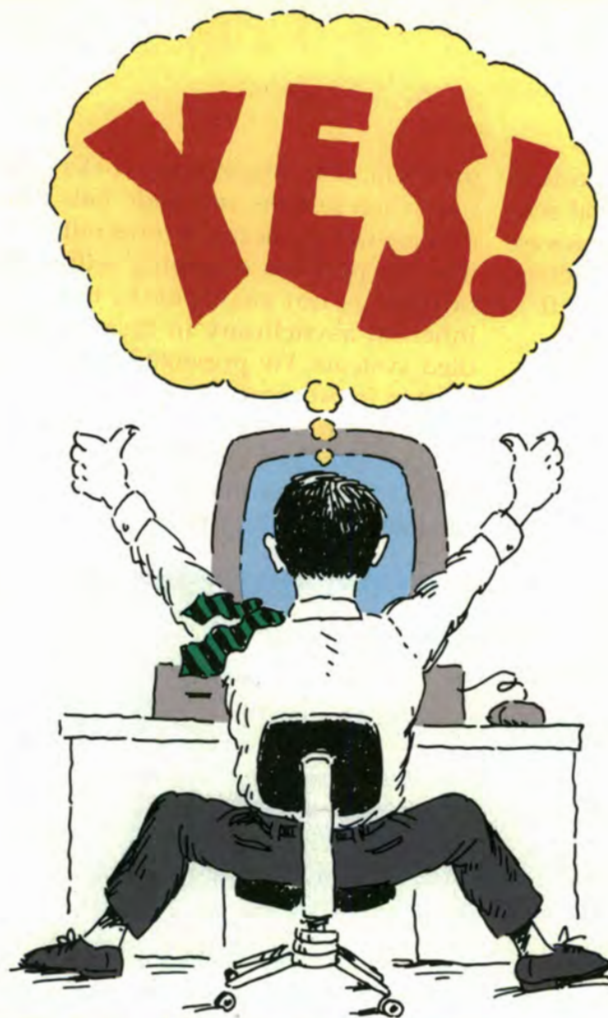
    oldtp.tv_sec = 0;
    oldtp.tv_usec = 0;
    for (index = 0; index <= 20; index++) {
        /* create lock file */
        lkresult = 0; i = 0;
        while ((i < 3) && (lkresult != LOCKF_FILE_AVAILABLE)) {
            process_lock(sfstem, &lkresult);
            if (lkresult != LOCKF_FILE_AVAILABLE) {
                printf("sleeping for one second \n");
                DosSleep(1000);
                i++;}
        } /* end while */
        if (i == 3) printf(" Failed to get lock after 3 retries \n");
        else { /* open SF file */
            if ((fd = open( sfname, O_CREAT|O_RDWR, S_IREAD|S_IWRITE )) < 0)
                printf ("OPEN Error1\n"), exit(1);
            /* < ... use the SF ... > */
            close(fd);
            process_unlock(sfstem);} /* delete the lockfile */
        oldtp = tp;
    } /* end of for */
}
```

Figure 3. A test program for the algorithm

that the name of the resource be passed in as a parameter. Also before using the algorithm, one should pass in a maximum-resource-usage-time value.

The second file, `tstpprlock.c`, has a driver main program that exercises the lock and unlock functions provided in `pprlock.c`. By running two or more instances of this program, you can see how

contention for a common resource is resolved by the lock implementation. Several optimizations are possible in the given implementation. For example, you could build a common function to replace `process_lock1` and `process_lock2`. Users can also tailor this implementation to suit their needs and either make the algorithm even more leak-proof by



HARNESS THE POWER OF OS/2

INTRODUCING OS/2 MAGAZINE



Make the most of your PC with *OS/2 Magazine*! Every issue brings you the information you need to achieve maximum performance. From your hardware and software, to your operating system and peripherals.

OS/2 Magazine helps you harness the power of your PC. Why not flip open the next issue to timely feature articles like these:

- Common OS/2 problems and how to solve them
- How to detect OS/2 viruses—and keep your data safe
- Putting your OS/2 PC onto a local-area network
- Making multimedia applications come alive under OS/2
- Running Windows applications under the Workplace Shell

You'll also get hands-on tutorials. Expert technical advice. And unequivocally, unbiased product reviews—on HP LaserJets, S3 graphic boards and Intel fax modems, to name a few. Plus the latest news on OS/2, invaluable network support and more.

SEE FOR YOURSELF

First issue FREE! Check out the next issue of *OS/2 Magazine*. It's on us. If you like it, you can get a full year's worth (11 additional issues) for the low charter rate of \$29.95. You'll save 25% off the regular subscription rate!

Why not return the attached form today? You have so much to gain, and *absolutely* nothing to lose!

**SPECIAL
INTRODUCTORY
OFFER!**



YES! Send me a complimentary copy of *OS/2 Magazine*. If I like it, I can get a full year's worth (11 additional issues) for the low charter rate of \$29.95. I save 25% off what regular subscribers pay. If *OS/2 Magazine* is not for me, I'll simply write "cancel" on your invoice and that will be that. I owe *nothing*. I lose *nothing*. And the FREE issue is mine to keep.

NAME _____

ADDRESS _____

CITY/STATE/ZIP _____

Mail your order to: *OS/2 Magazine*, P.O. Box 56664, Boulder, CO 80323-6664, or call (800) 765-1291, or Fax (415) 905-2233.

Allow up to 6-8 weeks for delivery of your first issue. Canadian/Mexican subscribers please add \$10 for surface mail; overseas add \$40 for airmail. Prepayment drawn in U.S. dollars on a U.S. bank for foreign orders.



running the lock removal portion of the algorithm in a critical section or relax it by doing away with `process_lock2` calls altogether and implementing Algorithm II.

SUMMARY

In this article, we discussed the problem of mutual exclusion while accessing a shared file using NFS on OS/2 and presented a simple algorithm for solving the

problem. The algorithm works fine when process and node failures are not a concern. It turns out that the problem of coping with failures is not easy due to the inherent asynchrony in distributed systems. We presented step-by-step improvements to the simple algorithm, which gradually increase the algorithm's reliability. Finally, we presented a C implementation of Algorithm III which

subsumes all of the algorithms discussed in the article.

ACKNOWLEDGMENTS

I thank Kamaal Thadani for implementing the various previous versions of the locking algorithm and also for pointing out examples in which the algorithms fail. I also thank Ira Forman for pointing out counter examples that led to improvements in Algorithm III.

REFERENCES

TCP/IP Version 1.2.1 for OS/2(Refresh): User Guide, Chapter 15, "Using Network File System Client." IBM publication number SC31-6076-03.

TCP/IP Version 1.2.1 for OS/2(Refresh): Installation and Maintenance, Chapters 8 and 9. IBM publication number SC31-6075-03.

SOMObjects Developer Toolkit: User Guide, Version 2.0, June 1993. Order by calling 1-800-342-6672 (USA), 1-800-465-7999 (Canada).

Hari Madduri received a Ph.D. in computer science from the University of Wisconsin-Madison in 1985, worked at Honeywell (Research) for five years, and joined IBM in 1990. He has been working on various aspects of distributed systems for over 10 years, including algorithms, fault tolerance, tool support, and object-oriented design. He is currently working in the Object Technology Products group in IBM Austin, building SOM-based class libraries and frameworks.

The Path To A Good GUI Isn't Always Clear

Navigate the maze with the NEW 2.1 Professional Developers ToolKit

Promises are cheap, GUI development tools aren't. Gpf 2.1 demo software is **FREE**. Try Gpf before you buy any tool and you'll agree that **Gpf 2.1 DELIVERS**.

With this powerful point and click visual programming environment you can create a CUA '91 Graphical User Interface for **OS/2 Presentation Manager®** or **MS DOS/Windows®** in as little as 10% of the time required to hand code the same design.

What You See Is What You Get programming reduces weeks of coding to hours. Quickly prototype and test your interface, then let **Gpf write the code**. Gpf generates error free, structured Object Oriented C or C++ source, complete with embedded SQL for OS/2 DataBase. Custom Help and Logic are added to controls as they are drawn to create full **Client/Server** or **Stand-alone** applications!

Gpf is hosted on OS/2 2.x and generates **native code** for OS/2 2.x, OS/2 1.3.x and MS Windows 3.0 or 3.1 Of course this means maximum performance with **no run time module** and **no royalties!**

Seeing Gpf is believing!

- Design 32 or 16 bit GUIs for IBM OS/2® or Microsoft Windows®
- Be creative, WYSIWYG design environment
- Full application and/or DLL generation
- Minimal time to application delivery
- Full CUA '91 Control set, 32 or 16 bit
- Reusable Objects
- Automatic documentation

Circle Reader Service Number 15

Gpf

Try us, **FREE** Demo Software Available
Order Gpf Pro ToolKit today for just \$1440⁰⁰
Separately: Gpf 2.1 for \$1295⁰⁰ & GpfTools for \$195⁰⁰
Or, try: Gpf 2.0 Single Platform, now available for just \$495⁰⁰

Call Gpf Systems Inc. at: (800) 831-0017



* GUI Programming Facility



SYSTEMS, INC. Phone (203) 873-3300 • fax (203) 873-3302 30 Falls Road, Moodus, CT 06469



This article, the second in a series on LAN NetView, explains how to integrate a managing application with LAN NetView's View user interface component. It also shows how to integrate managing applications on the View component's topology display, enabling users to share displayed objects.

By CHUCK MCKELLEY and JACOB M. RZEPKA

Using the LAN NetView Basic User Interface APIs

Now that you've learned how to communicate between managing and managed stations (see "Introduction to LAN NetView Application Development: An XMP/XOM Primer," November/December 1993), it's time to learn how to integrate your managing application with the LAN NetView program's View user interface component. The View component allows the system administrator to navigate through the network topology using an object-oriented, workplace shell-like interface.

In addition, you can integrate managing applications on the View component's topology display, which enables them to share the displayed objects. This allows the system administrator to navigate to the object of interest and select the application action to be performed against it in the same manner, no matter how many applications are integrated with the topology display.

The View component relieves the application programmer from the burden of supplying the navigation logic. It also allows the management program to retrieve topology and user interface information from common databases using OS/2's System Object Model (SOM).

THE TOPOLOGY/DISCOVERY SERVICE

Before you can learn how to integrate a managing application with the View topology display, you must understand how the system administrator uses it. The View display reflects the contents of the LAN NetView platform's discovery database. The discovery database contains information that LAN NetView discovery processes have discovered about the resources in the network. This information is arranged in three hierarchical levels.

The top level object is the physical system that represents a managed station. Either the media access control (MAC) address of the primary communication adaptor or an Internet protocol (IP) address names the physical system in the discovery database.

Physical systems contain the next level objects called logical systems. A logical system is a view of the physical system as seen by a management agent. For example, the LAN NetView program's system agent implements an International Standards Organization (ISO) system object that provides a Common Management Information Protocol (CMIP) view of the system. A Simple Network Management Protocol (SNMP) agent sees the system as an IP address with an SNMP system group.



Chuck McKelley



Jacob M. Rzepka



A Novell NetWare agent sees a system differently from either an SNMP agent or a CMIP agent. Each of these views appears as a logical system in the LAN NetView platform's discovery database. All three of them could be present in the same physical system.

Logical systems contain third-level objects known as resources. The discovery database does not contain information on all the resources on the managed stations. For CMIP logical systems, the database contains all of the information on each object that is named under the ISO system object in the naming tree, is no more than one level below the system object, and contains state or status attributes.

For SNMP logical systems, it contains the information from the SNMP system group. Only three levels of objects—physical sys-

tems, logical systems, and resources—and their associated attributes are contained in the discovery database and presented to the system administrator in the View topology display.

USER INTERFACE MODEL

Using the View component's user interface, the system administrator sees the topology display as nested container objects similar to the OS/2 workplace shell directory tree.

The top-level container is the management collection. A management collection is simply a set of systems that the system administrator has grouped together because they share something in common.

For example, the systems may all be in one building, or they may all contain LAN NetView agents for IBM's LAN Server. When the system administrator

opens a management collection, its contents are displayed. The contents of a management collection are systems.

In this context, systems are shadows of the physical systems contained in the discovery database. This is due to the fact that a given physical system may be contained in multiple management collections. The contents of a system container are resources. Both logical systems and discovery-database resources appear as resources in the View resource display.

PROGRAMMING CONCEPTS

Like the OS/2 workplace shell, the View component of the LAN NetView program was developed using object-oriented programming and OS/2's System Object Model. Your management application program must also be developed using object-oriented

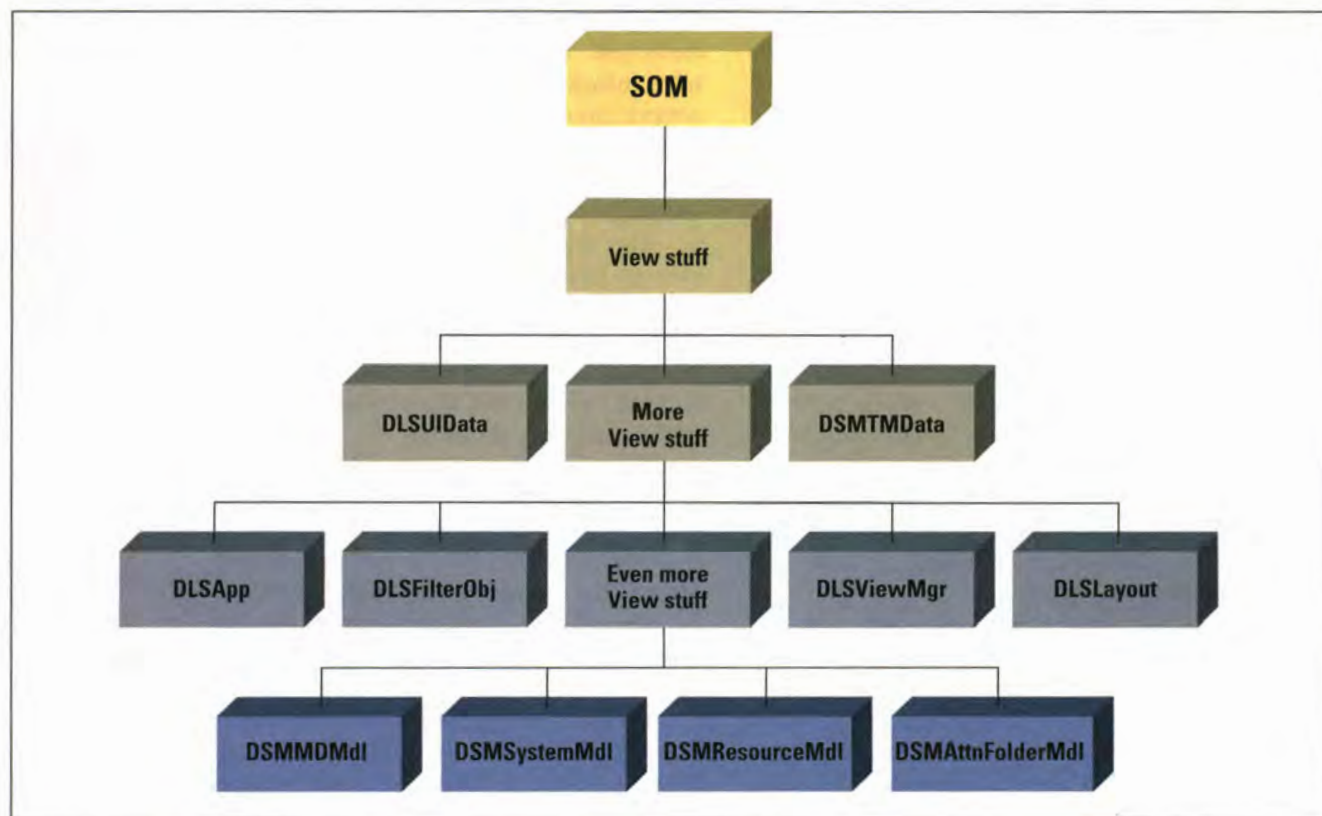
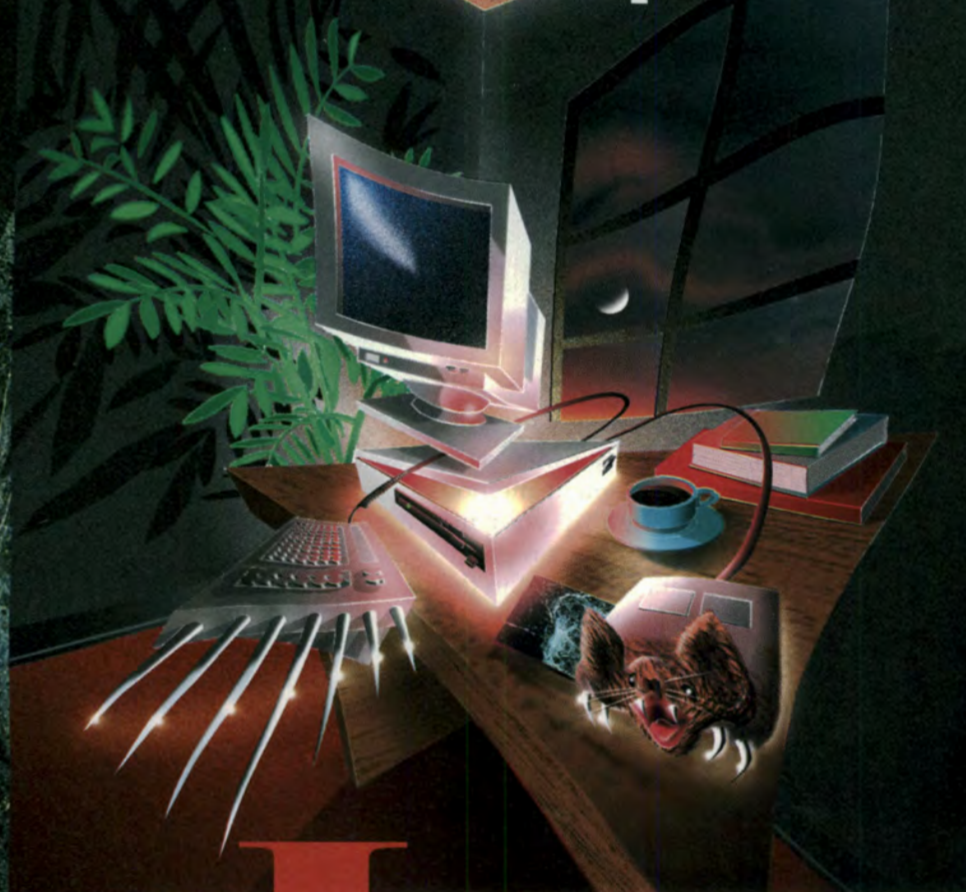


Figure 1. Hierarchy diagram

Client/Server 4GLs. Open as Pandora's Box.



In today's high technology marketplace, you hear a lot of stories about proprietary 4GLs for client/server application development. Stories that lure you ... that make 4GLs sound flexible, fast and open. Beware the paradox: how can 4GLs be open when someone else holds the key? Using client/server 4GL tools means opening Pandora's box of proprietary languages, vendor lock-in, and runtimes up to 16 times slower. With surprises like these, trusting your business-critical applications to 4GLs should scare you half to death.

With KASE:VIP, you get client/server solutions in standard languages like C, C++ and COBOL. Safe, secure solutions that leverage your current investment instead of replacing it. Using interactive visual design and code generation technology, KASE:VIP delivers the rapid development and increased productivity promised by 4GLs— but without unpleasant surprises.



Don't open Pandora's box. Get the facts in our free white paper, "No Strings, Boxes, or Borrowed Time: Overcoming 4GL Dependencies with Standard Language Client/Server Development Tools."

Just call 1 (800) 888-4335 or (404) 448-4240.

The leader in standard language-based client/server tools.

KASEWORKS™
△△△△△ Enabling Client/Server Strategies

Fact. Not fantasy.



programming and SOM. While this article is not intended to teach you object-oriented programming, it is useful to define a few terms as we will use them here:

- Classes are the definitions of program objects in terms of the data they contain and the program logic that operates on the data.
- Objects are particular instances of some class. An object has unique values assigned to the data items defined in the class. Assume, for example, that `MyClass` defines a data item called `Color`. The value of `Color` in instance `A` of `MyClass` might be `Red`, while the value of `Color` in instance `B` of `MyClass` might be `Blue`.
- Methods are simply procedures defined in the class that perform actions, usually associated with the data items.

A class, `A`, may be defined in such a way that it inherits properties of another class, `B`. This is known as subclassing. We say that class `A` is a subclass or child class of `B`, and `B` is the parent class of `A`.

When a class `A` is a subclass of class `B`, it inherits all the properties of the parent class. If class `A` must have some different properties than `B`, it can implement them by adding methods to those inherited from the parent class or by changing the methods it has inherited. This latter process is known as overriding a method from the parent class.

The process of subclassing parent classes and overriding their methods soon results in a hierarchy of classes, which is known as the class hierarchy.

PROGRAMMING MODEL

In the View component's object-oriented programming model, the View container objects (that is, management collections, sys-

```
MyMenuItem(pHier,pModel,pAppParm,pSelected)
{
    Initialize variables and get the system record handle;
    Get the icon title text;
    Check for the presence of an SNMP agent;
    Check for the presence of a CMIP OS/2 agent;
    Show the user which agents exist and call the
        XMP interface program to display data if requested;
    Cleanup buffers;
} // end MyMenuItem
```

Figure 2. Sample application pseudocode

tems, and resources), are modeled as program objects. These objects, usually called associated objects, implement methods that may be called by your management application program to act on the data contained in them.

Management application programs are also program objects in the View model. You implement your management application program by developing an application program class that contains methods that the View component's associated objects may call. You use SOM to define the class that describes your application object and then use C to add logic to the methods in your management application program class.

To make it easier for you to develop your management application program in this manner, the View class hierarchy contains a stub application class you can subclass to create your application program. This class is the `DLSApp` class in the View class hierarchy diagram illustrated in Figure 1.

In addition to the associated object classes and the application stub class, View classes also supply an object to access the discovery database information and access and update the user interface information stored in the View user interface database.

DEVELOPING AN APPLICATION CLASS

The process of developing an application class consists of five steps:

- Create a SOM class definition (.CSC) file. This is a source file that specifies the parent of your applications class, the class name, new or overridden methods, and any data items that instances of your class own.
- Compile the .CSC file using the SOM compiler `SC.EXE`. This step produces a skeleton .C file that contains your class definition and a function declaration for each method that is introduced or overridden in your class. It also produces several include files.
- Use the C compiler and OS/2 linker to create a dynamic link library (DLL) containing your class.
- Use the resource compiler `RC.EXE` to add any necessary Presentation Manager resources to your DLL.
- Use the View component's registration command `LNVREG` to register your class with the View registration services.

That's all there is to it. To show how easy it is, let's develop a View-based front end for the X/Open Management Protocol (XMP) application we introduced in the last article, "Introduction to LAN NetView Application Development: An

At this

So are lava lamps and bell-bottoms.

The theme of our conference is "retro," but the technologies are up-to-the-minute.

TECHNICAL That's why software designers, developers, technical coordinators, device driver developers, LAN specialists, MIS managers, consultants and training executives definitely shouldn't miss it.

Improve your productivity with sessions on OS/2,[®] LAN systems, graphics, OOP, multimedia, pen, database and communications. Learn to write device drivers for displays, printers, storage, LANs and input devices.

bugs are



software conference,

Everyone is invited to experiment in our OS/2 and LAN lab, and see exhibits from

See the latest PSP technologies, April 25-29, 1994
Now including the Device Driver Conference

software vendors who exploit PSP products.

Hear from IBM's own Lee Reiswig, Jr., President of the Personal Software Products Division.

cool.

Also, enjoy premiums, raffles, a "special event" and more. Register before March 21, 1994 and save \$100 on the \$895

fee. For more information or to sign up, call 1 800 872-7109 (USA and Canada).*

It's gonna be groovy.

Operate at a higher level.™

*Outside the USA and Canada, call 1 508 443-4990. Volkswagen, the Volkswagen logo and "Beetle" are registered trademarks of Volkswagen, AG. Used by permission of Volkswagen, AG. IBM and OS/2 are registered trademarks and "Operate at a higher level" is a trademark of International Business Machines Corporation. © 1993 IBM Corp.



XMP/XOM Primer," November/December 1993.

Our XMP program used the IP address for SNMP agents or the distinguished name for CMIP agents to access the agent to retrieve or modify data. We used a hand-built file containing the IP addresses and distinguished names, which we needed to address the agents. In this example, we obtain that information from the View services and let the View display provide the functions necessary to select the system of interest.

The overall flow will work like this:

- The LAN NetView discovery services will discover systems and resources in our network and save the information in the discovery database.
- The View user interface services will display the network topology and allow the system administrator to select a system. The View services will also allow the user to select our program from a popup menu associated with the selected system.
- When the user selects our menu, the View services will create an instance of our application class, called an application object, that is associated with the selected system. Then, it will call a menu item method in our application object.
- When our menu item method receives control, it will use View services to access the discovery database object to obtain a record handle for the system. We will use this item later to retrieve information from the discovery database.

Next, our method will use the View component's user interface database services to obtain the icon text for the selected system.

Then, it will use View services to access the discovery data-

```
-- dsmsamp1.csc
include <duiapobj.sc>
class: DSMSamp1,
    global;

parent: DLSApp;

passthru: C.h, before;
#define INCL_WIN
#define INCL_BASE
#define INCL_ERRORS
#define INCL_DOSPROCESS
#include <os2.h>
#include <string.h>
#include <stdio.h>
#include <dsmdef.h>
#include <duicmudt.h>
#include <dsmcmtmd.h>
#include "dsmsmp01.h"
#include "dmoh.h"
endpassthru;

data:
    HMODULE ModHandle;

methods:

    int MyMenuItem1(StringArrayObj *pHier,
                    SOMany *pModel,
                    PVOID pAppParm,
                    SOMany *pSelected);
    -- Retrieve CMIP or SNMP information from a selected system
    ULONG CMIPReq(REQUESTSTUFF *pCMIPReqParms);
    -- This method handles the displays for the case where
    -- there is an OS/2 CMIP agent but there is not an SNMP agent.

    ULONG SNMPReq(REQUESTSTUFF *pSNMPReqParms);
    -- This method handles the displays for the case where
    -- there is an SNMP agent but there is not an OS/2 CMIP agent.
    ULONG CMIPSNMPReq(REQUESTSTUFF *pCMIPReqParms,
                      REQUESTSTUFF *pSNMPReqParms);
    -- This method handles the displays for the case where
    -- there is both an SNMP agent and an OS/2 CMIP agent.
    override somInit;
    -- Instance setup
```

Figure 3. Sample .CSC file



"What happened to our changes?"

"Imagine the Marx Bros. doing team development on the LAN!"

Without automated software configuration management on the LAN, your client/server development efforts may resemble a comedy of errors. Consider the potential for disaster with multiple developers working in parallel on the same application at the same time.

PVCS ENDS CHAOS

The solution is INTERSOLV PVCS, a complete family of Software Configuration Management (SCM) solutions designed for the new world of client/server development. PVCS is the standard in SCM, working

identically across 17 operating environments with all types of development.

OVER 135,000 USERS

With over 135,000 users at 9,500 sites worldwide, PVCS is used by more developers than all other SCM tools combined! INTERSOLV PVCS accelerates delivery, improves quality, and leverages overall productivity by taking the chaos out of team development on the LAN.

To find out how PVCS can leverage your development goals call for seminar details or our information package.

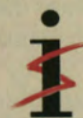
INTER SOLV PVCS leverages team productivity of LAN based developers by automating software configuration management. PVCS is a component of the INTERSOLV Development Suite, a family of development tools used individually to fill tactical gaps, or in combination to form a complete client/server development environment.

INTER SOLV Inc., 1700 NW 167th Pl., Beaverton, OR 97006.

FREE details on automating Software Configuration Management on the LAN.



**Call 800-547-PVCS
EXT. 500 ... TODAY!**



INTER SOLV

Your Edge In
Software Development

PVCS see what's new at SD '94 Booth #1637 San Jose, March 15-17

Circle Reader Service Number 19



base to determine if an SNMP agent or a LAN NetView system agent exists on the system. Our program will display an appropriate menu depending on whether it finds one, both, or none of these items. If the user decides to retrieve SNMP or CMIP information from the system, our program will call the XMP sample program to retrieve the data. Finally, the method will display the results of the XMP operations. Figure 2 shows the pseudocode for this function.

The first step in our sample program development is to design the PM screens and create the resources. We'll assume here that this step has already been done, so we will move on to creating our .CSC file, shown in Figure 3.

CREATING THE .CSC FILE

The parent class of our application class will be DLSApp, so we must specify that in our .CSC file. We must also include the SOM language-independent definition of DLSApp, which is in the DUIAPOBJ.SC file. We will introduce one new menu item method, MyMenuItem1, so we must specify that in our .CSC file also. Our .CSC file also includes an override of the somInit method and three other new methods that are used to present the PM screens for our program.

Another thing we can specify in the .CSC is information that we want SOM to pass to the include files it generates. This information is in the passthru section. In our .CSC, we have told SOM to pass through includes for some standard .H files and two .H files that contain the definitions of our private data structures, PM definitions, and so on.

Since menu item methods must conform to a specific syntax, the prototype for a menu item is

```
SOM_Scope int  SOMLINK MyMenuItem1(DSMSamp1 *somSelf,
                                     STRING_ARRAY_OBJ *pHier,
                                     SOMAny *pModel,
                                     PVOID pAppParm,
                                     SOMAny *pSelected)
{
    ULONG      uLRC;           // Return code
    char        *Names[10];    // Name pointer array
    PSZ         IconText;      // Icon text
    PSZ         SystemName;    // Displayable system name
    DLSUIData   *pUIDB;        // UI DB pointer
    UIDATA_KEYS Keys;          // UI database key struct
    UIDATA_INFO UIInfo;        // UI info struct
    DSMTMData   *pTDB;        // Topology DB pointer
    char        **LongNames;   // Long name array
    HTDBRECS    *pHandles;     // Handle array ptr
    HTDBREC     *pThisSys;     // System handle pointer
    TMD_ATTRS   *pAttributes;  // Attribute array pointer
    TMD_ATTR    *pAttr;        // Attribute pointer
    TDBENTS     *pResources;   // Resource array pointer
    PSZ         AttrValue;     // Attribute value
    PSZ         IPAddr;        // SNMP IP address
    int         ResourceCount;  // Number of resources
    pTOP_DN     SysObjectDN;   // CMIP sys obj DN
    TOP_DN_BUF  *pDNArray;     // From QueryDN
    HAB         hab;           // PM Anchor block handle
    HMQ         hmq;           // PM Message queue handle
    REQUESTSTUFF CMIPReqParms; // CMIP request parameters
    REQUESTSTUFF SNMPReqParms; // SNMP request parameters
}
```

Figure 4. Method declaration and variables

included in the DUIAPOBJ.SC file. This prototype does not represent a real method and is not intended to be overridden.

Let's take a look at the parameter list of our menu item method. The first parameter of any SOM method call is a pointer to the object receiving the call. This is never specified in a .CSC file, however, so we have not specified it here. The first parameter in our method prototype is pHier. It is defined as a pointer to a STRING_ARRAY_OBJ structure. This structure contains the string representation of the View class and instance names of the associated object that called the menu item method. STRING_ARRAY_OBJ struc-

tures are used extensively in View methods. They contain a pointer to Array, which is an array of pointers to null-terminated strings, and an integer field, Count, which is the number of string pointers in the array. You refer to the class name of the associated object, which is the first string in the STRING_ARRAY_OBJ pointed to by pHier, like this:

```
MyStringPointer = pHier->Array[0];
```

The second parameter, pModel, is a pointer to the associated object itself. This is the object whose class and instance names are contained in the STRING_ARRAY_OBJ structure.


```

SystemName = (PSZ) NULL;
IPAddr = (PSZ) NULL;
SysObjectDN = (pTOP_DN) NULL;
pDNArray = (TOP_DN_BUF *) NULL;
//
// Get topology database pointer
//
pTDB = get_pTOPDBM(_DSMTMDData);
//
// Get system record handle
//
LongNames = Names; // Point to my name array space
LongNames[0] = pHier->Array[1]; // Get system long name pHandles =
_GetTDBHandles(pTDB, LongNames, 1); // Get handle array
pThisSys = (* pHandles) [0]; // Extract handle
if (pThisSys == NULL) { // Bail out if no handle
return (int) 0;
} /* endif */

```

Figure 5. Initialize and retrieve system record handle

The third parameter is a pointer to a data item you can specify at the time the View registration service registers the menu item method. Although this item is defined as a pointer to void, the registration service only allows strings to be registered. This item is useful, if you use the same method for several menu items. You can use the registration parameter with a switch statement to determine the context of the method call.

The last parameter, *pSelected*, is a pointer to a View object that contains the list of selected items for the associated object.

After we execute the SOM compiler *SC.EXE* against our *.CSC* file, we will have a skeleton *.C* file that contains a function declara-



Now You Can Easily Backup, Restore, Migrate and Manage Your Workplace Shell Desktop With ...



DeskMan/2™ is the first and only tool designed to manage the Workplace Shell™!

DeskMan/2 for OS/2 is unmatched in its ability to fully exploit the power and versatility of OS/2! Managers can use its security features to protect access to windows or objects. Administrators can easily and quickly standardize a department's desktops for maximum job efficiency. Individual users can quickly and easily migrate icons, objects and desktops between office and home machines. Or use **DeskMan/2** to backup and completely restore a custom desktop.

Incorporating the powerful **VUEMan/2™** virtual desktop & window manager; the potent new WPS snapshot facility; and more, **DeskMan/2** is packed with features!

"I installed OS/2® 2.1 golden code, and **DeskMan/2** saved me at least 4 hours of tedious labor. I can't imagine being without it."

David Barnes - Senior Staff member - IBM's OS/2 Executive Briefing Center

- | | |
|---|---|
| ☒ API & GUI for local & remote access | ☒ Change multiple folders or icons all at once - just drag and drop |
| ☒ CID enabled & LAD/2 supported | ☒ Full administrative control of all features |
| ☒ Delete "undeletable" objects | ☒ Restore, save, migrate any/all objects |
| ☒ Prevent the deletion of any object | ☒ Total management of Workplace Shell |
| ☒ Query any object for settings | ☒ And much more ... |
| ☒ Secure individual windows | ☒ Call DevTech for current info! |
| ☒ Secure groups of windows | |

DevTech

Development Technologies, Inc.
Software Development & Technology Transfer
308 Springwood Road, Forest Acres, SC 29206-2113 Voice: (803) 790-9230 Fax: (803) 738-0218



© Copyright 1994 Development Technologies, Inc. All rights reserved. DeskMan, DeskMan/2 and VUEMan/2 are trademarks of Development Technologies, Inc. Workplace Shell is a trademark of International Business Machines Corporation. IBM and OS/2 are registered trademarks of International Business Machines Corporation.



ration for our menu item method. The next step is to use C to add logic to our .C file. Figure 4 shows the method declaration for `MyMenuItem1` and the variables used by the method.

RETRIEVING THE SYSTEM RECORD HANDLE

Figure 5 shows the first block of code in our method. This code initializes some variables that we will test later in the program and accesses the discovery database to retrieve the system record handle. Record handles are the discovery database object's private keys.

To retrieve information from the discovery database, we use the discovery database class `DSMTMDData`. This class has a single instance allocated by the View services at initialization time. This instance must be used for all calls to the class. We can use the `SOM get_` macro to retrieve its address, as shown in the example. The methods in the `DSMTMDData` class use some special structures that are defined in the `DSMCMTMD.H` file shipped with the LAN NetView product.

The discovery database object retrieves information regarding physical systems and resources and their respective attributes. In this context, discovery database logical systems are considered resources. To get the record handle for our system, we must call the `DSMTMDData` class's `GetTDBHandles` method. The input to this method is an array of long names and an integer that indicates the number of elements in the array.

The return information is a pointer to an array of record handles, one for each long name in the input array. If any long name does not have a corresponding record handle, the handle for that position in the output array will be set to `NULL`. In our case, we will pass an array with one element:

```
//
//      Get user interface database pointer
//
pUIDB = get_pUDT(_DLSUIData);
//
//      Set up user interface database interface structure
//
UIInfo.UIDataKeys = &Keys;          // Set key struct address
//
//      Set up key structure
/[
Keys.Class = pHier->Array[0];        // Get class name
Keys.Name = pHier->Array[1];         // Get instance (long) name
Keys.Datatype = DTYPE_TITLE;        // Set data type to icon title
Keys.Subtype1 = (PVOID) KEY_TERM;    // Set terminator
//
//      Call UI database method
//
IconText = _QueryData(pUIDB,&UIInfo);
if (IconText != (PSZ) NULL) {
    //
    //      We have icon text so save it in a private buffer
    //
    SystemName = (PSZ) _AllocBuffer(somSelf,strlen(IconText) + 1);
    strcpy(SystemName,IconText);
} else {
    //
    //      Call discovery database for short name if no icon text
    //
    pAttributes =
        _QueryAttribute(pTDB,pThisSys,TMD_ATTR_SNAME,(char *) NULL);
    if (pAttributes != NULL) {
        //
        //      Extract short name and save it
        //
        pAttr = &(*pAttributes)[0];    // Get attribute struc pointer
        IconText = (PSZ) pAttr->valueList; // Get attribute address
        SystemName = (PSZ) _AllocBuffer(somSelf,strlen(IconText) + 1);
        strcpy(SystemName,IconText);
    } else {
        //
        //      No short name? Just use long name
        //
        SystemName =
            (PSZ) _AllocBuffer(somSelf,strlen(pHier->Array[1]));
        strcpy(SystemName,pHier->Array[1]);
    } /* endif */
    //
    //      Free attribute array buffer
    //
    _FreeBuffer(pTDB,(char *) pAttributes);
} /* endif */
```

Figure 6. Retrieving the icon text

the second element of the array pointed to by `pHier`.

RETRIEVING THE ICON TEXT

Our program's next task is to determine the text that is displayed under the icon for our associated system object. Figure 6 shows the code block for this. The icon text is normally an attribute of the system object called the short name. The short name attribute is stored in the discovery database. However, the system administrator can change the text that appears under the icon. If this occurs, the name of the system doesn't change, but the text entered by the system administrator is displayed under the icon and saved in the View user interface database. To determine the name displayed under the icon,

we must first check the user interface database to see if the system administrator has changed the text. If the text has not been changed, we must retrieve the short name from the discovery database.

We will use the `QueryData` method of the user interface database class `DLSUIData` to retrieve the icon text. Before we call the `QueryData` method, however, we must obtain the address of an instance of `DLSUIData`. As in the case of the discovery database class, the View services create an instance of `DLSUIData` when it is first initialized. Only this one instance should be used to interface with the user interface database. We can obtain its address with the `SOM get_` macro, as shown in the example.

To retrieve data from the user interface database, we must first fill in a `UIDATA_KEYS` structure with a key that will access the data. Data in the user interface database is located by a compound key with six fields:

- **Class** is the name of the View class for the object that owns the data. In our case, it is the name of the View `System` class passed as the first array element pointed to by `pHier`.
- **Name** is the View instance name, commonly known as the long name. Our system's long name is the second array element `pHier` points to.
- **Datatype** is one of the data types defined in the `DSMDEF.H` file. The data type for icon text is `DTYPE_TITLE`.
- **Subtype1** is an unused qualifier.



Next issue of

**OS/2
Developer**
THE MAGAZINE FOR ADVANCED SOFTWARE DEVELOPMENT

will feature...

tools and techniques to
help you develop
MULTIMEDIA
applications for OS/2

May/June OS/2 DEVELOPER covers:

- Adding sound to OS/2 applications
- Creating multimedia applications that are responsive to users
- Using OS/2 device drivers
- Multimedia objects for OS/2 applications
- A Buyer's Guide to multimedia tools and utilities

Also, GUI column debuts
in the May/June issue!

Take control of your OS/2 scheduling needs with

ATS

Version 2.0b

Advanced Task Scheduler for OS/2

Are you looking for a way to manage your production job streams? Do you have programs that need to run after the completion of one or more other programs? Do you want to process files that have just been received from another system or modified locally? Do you need to schedule around holidays and periods of heavy CPU load?

With ATS, you can accomplish all of this and MUCH MORE. With ATS you can run your programs when YOU want them to run and you don't even have to be there to start or monitor them.

Here are some of the features of ATS for OS/2:

- * Create True Job Streams
- * Programs can be scheduled to run anytime.
- * Programs can be dependent upon Files, Other Programs, Holiday Schedules, and External Signals
- * Complete Logging - To File and Console
- * Console Display of Running Tasks
- * Console Display of Holidays, Events and Tasks
- * Programming Interface for External Signals
- * Only \$249.00

MHR

Software and Consulting

2227 U.S. Highway #1 * Suite 146 * North Brunswick, NJ 08902
Voice: (908) 821 - 0359 Fax: (908) 821-0350

Circle Reader Service Number 21



- Subtype2 is an unused qualifier.
- Subtype3 is an unused qualifier.

The first unused field in our example, Subtype1, must be set to the constant, KEY_TERM, which is defined in DSMDEF.H. We must build the key in a UIDATA_KEYS structure, which our class provides. In addition, we must store the address of the UIDATA_KEYS structure in the UIDataKeys field of a UIDATA_INFO structure, which our class provides. This structure will be input to the QueryData method of the user interface database class DLSUIData.

If QueryData returns a value other than NULL, the system administrator has assigned a custom label to this icon, and we can save that to use as the displayable system name. If a NULL pointer is returned, however, then the short name is the name under the icon, so we must retrieve it from the discovery database.

The short name is an attribute of a system or resource, so we will need to call the discovery database object's QueryAttribute method. This method, like most methods in the DSMTMDData class, requires a record handle as input.

We can use the record handle we retrieved during initialization when we issue the QueryAttribute method call to retrieve the short name. In addition to a record handle, this method requires either an integer attribute identifier or a string that is the dotted ASCII representation of the object identifier for a CMIP attribute.

The short name is one of the attributes defined in DSMCMTMD.H, so we can pass the integer attribute identifier. If the attribute does not exist, then QueryAttribute will return NULL. If it does exist, the method returns a pointer to an array of TMD_ATTR structures, each of which contains a pointer to the list of values of the attribute and an integer that specifies the num-

```
//
//      Call discovery database for SNMP SYSDescription MIB variable
//
pAttributes =
    _QueryAttribute(pTDB,pThisSys,TMD_ATTR_SNMPSYSDDESC,(char *) NULL);
if (pAttributes != NULL) {
    //
    //      Free attribute array buffer
    //
    _FreeBuffer(pTDB,(char *) pAttributes);
    //
    //      We have an SNMP agent so get the IP address
    //
    pAttributes =
        _QueryAttribute(pTDB,pThisSys,TMD_ATTR_IPADDRS,(char *) NULL);
    pAttr = &(*pAttributes)[0];    // Get attribute struc pointer
    AttrValue = (PSZ) pAttr->valueList; // Extract IP address
    IPAddr = (PSZ) _AllocBuffer(strlen(AttrValue) + 1);
    strcpy(IPAddr,AttrValue);    // Save IP address
    //
    //      Build the parameter structure for the PM and XMP calls
    //
    SNMPReqParms.SystemName = SystemName;
    strcpy(SNMPReqParms.HostName.SNMPName.IPAddress,IPAddr);
    strcpy(SNMPReqParms.HostName.SNMPName.CommunityName,"PUBLIC");
} /* endif */
//
//      Free attribute buffer
//
_FreeBuffer(pTDB,(char *) pAttributes);
```

Figure 7. Determining the presence of an SNMP agent

ber of attribute values in the list. The short name is a simple attribute with only one value, so there is only one element in the array and only one value in the list.

Since most applications need to retrieve the short name of a system object, the LAN NetView product provides an API, called GetShortName, that contains the logic we have just developed.

DETERMINING THE PRESENCE OF AGENTS

The last task we must perform before we display our first screen

is to determine if SNMP or LAN NetView agents exist on the system. We can use the DSMTMDData's QueryAttribute method to perform part of this task also. Figure 7 illustrates the block of code that determines if an SNMP agent exists.

To determine if an SNMP agent exists on the system and retrieve the IP address necessary to communicate with the agent, we can use DSMTMDData's QueryAttribute method to retrieve the value of the TMD_ATTR_SNMPSYSDDESC attribute. We can use the same record handle we retrieved in the initialization

PEN &

PORTABLE
COMPUTING
CONFERENCE
AND EXPOSITION

.....

BOSTON, MA
WORLD TRADE CENTER
MAY 2-5, 1994

EXPLORING
PORTABLE
COMPUTING,
MOBILE
CONNECTIVITY
AND
PEN TECHNOLOGY



For more information and
to receive a FREE brochure
call 508-649-4200

Sponsored By



METROPOLITAN COLLEGE
CENTER FOR INFORMATION TECHNOLOGY



routine. If this request returns a NULL pointer, no SNMP agent exists on the system. If the request is successful, we will know there is an SNMP agent, so we can use the `QueryAttribute` method again, this time to retrieve the IP address.

To check for a LAN NetView OS/2 agent and retrieve the necessary addressing information, we will use two new methods. Again, we will use the `DSMTMData` class and the system record handle from our initialization routine. Figure 8 illustrates the block of code that performs this function.

The first thing we must do is test for the presence of a LAN NetView OS/2 agent. This object is a resource. To test for its presence, we must use the `DSMTMData` class's `QueryResources` method. We can use this method to return a list of all the resources on a system; however, we will use it to test for the presence of a single resource.

As the example shows, the input to this method is a system record handle, the address of an integer field where `DSMTMData` can put the count of resources returned and a pointer to a character string representing the resource ID. If this last parameter is NULL, then all of the resources will be returned. If it is set to the string form of a CMIP object identifier, then only the data pertaining to that resource will be returned. If the resource doesn't exist, then a NULL is returned. The object identifiers for some of the more common resources are defined in `DSMDEF.H`. We will pass the object identifier for the OS2 object that is the top-level object in the LAN NetView OS/2 agent.

If our `QueryResources` call is successful, we will know that an OS/2 agent exists on the machine, but we will not know the address of the OS/2 object. To retrieve

```
//  
//      Call discovery database for OS/2 agent object  
//  
pResources =  
    _QueryResources(pTDB,pThisSys,&ResourceCount,OS2);  
if (pResources != NULL) {  
    //  
    //      Call discovery database for LAN NetView system object  
    //      distinguished name  
    //  
    pDNArray = _QueryDN(pTDB,pThisSys,TMD_DSM_SYS);  
    SysObjectDN = (*pDNArray) [0];  
    //  
    //      Build the parameter structure for the PM and XMP calls  
    //  
    CMIPReqParms.SystemName = SystemName;  
    CMIPReqParms.HostName.CMIPName.pEncodedDN = (PVOID) SysObjectDN;  
} /* endif */
```

Figure 8. Determining the existence of a CMIP OS/2 agent

```
//  
//      Free record handle buffer  
//  
_FreeBuffer(pTDB,(char *) pHandles);  
//  
//      Free icon title save buffer  
//  
if (SystemName != (PSZ) NULL) {  
    _FreeBuffer(somSelf,SystemName);  
} /* endif */  
//  
//      Free IP address save buffer  
//  
if (IPAddr != (PSZ) NULL) {  
    _FreeBuffer(somSelf,IPAddr);  
} /* endif */  
//  
//      Free distinguished name buffer  
//  
if (pDNArray != NULL) {  
    _FreeBuffer(pTDB,(char *) pDNArray);  
} /* endif */
```

Figure 9. Cleanup

A Blue-Ribbon Panel Named **POWER TRANSLATOR™ PROFESSIONAL** Most Innovative Software of 1993...



But our users are saying things that *really* make us proud.

The prize: 1993 Discover Award
for most innovative software.

The judges: John Dvorak,
Michael Miller, Marvin
Minsky, Esther Dyson, and
Stewart Cheifet. Their Affilia-
tions: PC Magazine, PC Computing,
MacUser, Computer World, Computer Chronicles

and MIT. The winner: **POWER TRANSLATOR
PROFESSIONAL** foreign language translation
software, from Globalink®!

But our users don't think about "innovation."
They have specific work in mind.

For Eric Vogt, translator for the American
Red Cross, **POWER TRANSLATOR PROFESSIONAL**
guarantees consistency and "...saves an enormous amount
of time... A person simply can't remember 5,000 scientific
terms and how they translated each of them the last time."
For Jamala Banks, a language student at Howard Univer-
sity, "what you would normally take a day on, you can

"...it saves an enormous amount of time..."

— Eric Vogt, American Red Cross translator

"...I really expanded my vocabulary, my grammar
...using the program has been just great."

— Jamala Banks, Howard University student

now virtually do in just minutes!"

Available for DOS, OS/2®,
Macintosh® and UNIX®, **POWER
TRANSLATOR PROFESSIONAL**
features speeds up to 100 times
faster than human translators,
easily-updated dictionaries of over
250,000 words and phrases and a 90%-plus
accuracy rate! Subject dictionaries are also available, in-
cluding computer, legal, business and finance.

Find out about the software package that's got Discover
Magazine – and a lot of other people – talking. Call
Globalink today and ask about **POWER TRANSLATOR
PROFESSIONAL**, your ticket to world-wide communication.



for OS/2



lobalink, Inc.

9302 Lee Highway, Fairfax, VA 22031
Int'l.: 1-703-273-5600
Fax: 1-703-273-3866

1-800-767-0035

Incredibly accurate translations: English to/from Spanish, French, or German

Circle Reader Service Number 23



this we must use the `QueryDN` method. This method requires a record handle as input and returns an X/Open Abstract Data Manipulation (XOM) encoded object, which is the distinguished name of the object associated with the record handle. If the record handle represents a system, then the method requires a second parameter to identify which logical system's distinguished name should be returned. We could ask for the distinguished name of the OS/2 object, since its record handle was returned in the previous `QueryResources` call. To allow our program to work better with our XMP example, however, we will ask for the distinguished name of the LAN NetView system object. To do that we must pass `TMD_DSM` as the system ID. The return is an array of distinguished name objects. In our case there is only one object in the array.

TERMINATION

The last thing our method must do is to deallocate any buffers that it has allocated or that the View objects have returned. Figure 9 shows this function.

REGISTERING AN APPLICATION CLASS

We are finished with the development of our menu item method. One more step remains before we can call it from the View component's topology display. We must create a registration file to register our class and menu item method with the View registration service.

The View registration file uses a simple tag language. Although there are many tags, we need only two for our registration file shown in Figure 10. We use the `APPLICATION` tag to register our application class; and we use the `ASSOFUNCTIONITEM` tag to register the

```
<
  APPLICATION
    CLASS=DSMSamp1
    MODULENAME=dsmsamp1.dll
    GLOBALINSTANCE=N
    NAME="LAN NetView Sample 1"
    SHORTNAME="DSM1"
    HELPPFILE=dsmsamp1.inf
  >
  ASSOFUNCTIONITEM
    TEXT="Get CMIP or SNMP data"
    CLASS=DSMSamp1
    ACCEL="C"
    METHOD=MyMenuItem1
    INFOTEXT="Get CMIP or SNMP data"
    HELPID=1
    ID=1
    PARENTID=ID_APP_ACTION
    ASSOCIATE="DSMSysMd1"
  >
```

Figure 10. Registration file

menu item method.

The syntax of all tags in the registration file is:

`<tag keyword=value ... keyword=value>`

There must be no spaces between the equal sign and either the keyword or the value.

The keywords for the `APPLICATION` tag in our example have the following meanings:

- **CLASS** is the name of our application class.
- **MODULENAME** is the name of the .DLL file that contains our class.
- **GLOBALINSTANCE** is the keyword that tells the View run-time services whether the application class can be instantiated more than once. The **N** means that our class can have as many instances as needed. A **Y** here means that our class works like a monolithic application and can only have one instance.
- **NAME** is the full name of our application.

- **SHORTNAME** is a short name for our application.
- **HELPPFILE** is the name of a help file that contains help information used when the user selects the help function when one of our menu items is selected.

The keywords on the `ASSOFUNCTIONITEM` tag have these meanings:

- **TEXT** is the text that appears on the menu.
- **CLASS** is the class to which this menu item belongs.
- **ACCEL** is a character in the menu item text that can be used to select our item from the keyboard.
- **METHOD** lists the name of our method.
- **INFOTEXT** contains the text that appears at the bottom of the window when our menu item is selected.
- **HELPID** is the ID of the item in the helpfile that pertains to our menu item.
- **ID** is an identifier for our menu item. If we had a cascaded menu,

we could use this identifier as an anchor for it. The View menu processing functions will ensure that there are no collisions between our menu item IDs and those of other applications.

- **PARENTID** is the menu ID of the menu that will contain our menu item. This ID must be one of our own or one of those listed in the DSMDEF.H file. We have specified that our menu item will appear on the Application Action menu.
- **ASSOCIATE** is the keyword that tells the registration service which classes of View objects should be associated with our menu item. We have specified that System objects should be associated with MyMenuItem1.

We can register our class and menu item by processing our registration file with the LNVREG command like this:

```
LNVRG dsmsamp1.reg
```

That's all there is to it. Our application is integrated with the LAN NetView topology display. The system administrator can use the View functions to navigate to the system of interest and execute our application. The only user interface code we had to write was the code that displays a simple dialogue and the information returned by our XMP application.

Chuck McKelley, IBM LAN Systems, Austin Programming Center, Austin, Tex., is an advisory programmer on the LAN NetView Vendor Support team. In his 26 years with IBM, McKelley has held various technical and management positions in field support and software development.

Jacob M. Rzepka, IBM Corp., Austin, Tex., is currently responsible for providing technical support to IBM customers porting or writing applications for the LAN Netview family of products. He joined IBM in Rochester, Minn., in 1974

where he was involved in the evaluation and testing of microprocessor-based optical and computer systems. In Austin, he was responsible for the development of communication products for 3270 emulation and was an architect for the IBM ES 1.0 Database Manager product. He received a B.S. in electrical engineer-

ing from Northwestern University in 1974 and an M.S. in computer science and control systems from the University of Minnesota in 1977. Rzepka has also completed post-graduate studies in the information systems and electrical engineering departments at the University of Texas in Austin.



The GammaTech Utilities are designed to enhance and complement your OS/2 system. These utilities provide the ability to perform vital maintenance and recovery operations easily without extensive technical knowledge. Use **GammaTech Utilities** for your critical data or you might lose it.

USE IT

GammaTech Utilities or for OS/2

- Undelete Erased Files
- Optimize HPFS and FAT Volumes
- Backup INI and Desktop Files
- Recover HPFS Volumes
- Protect and Backup Boot Sectors
- Identify and Mark Bad Sectors
- Reconstruct Boot Sectors
- Protect/Lock Files
- Permanently Erase Sensitive Data
- Mass Delete Selected Files
- Sort FAT Directories
- View and Edit Selected Files
- Disk Sector Edit (ASCII or Hex)
- Display Volume Information
- Alter File Attributes
- Add Comments to Files
- Display File Fragmentation
- Display Directory Information
- SAA/CUA Compliant PM Interface

For OS/2 Version 2

VISA, MasterCard, American Express, C.O.D. • Overnight Service Available

(405) 947-8080 • Fax (405) 632-6537

SofTouch Systems, Inc., Workstation Division

1300 S Meridian, Suite 600, Oklahoma City, Oklahoma 73108



*In this article, we will develop and debug a management system that uses the LAN NetView network management platform. The remote agents that will be a part of our management system will use the industry standard SNMP. **By STACEY TAYLOR***

Network Management: Using SNMP for LAN NetView

Have you ever had a problem with your workstation's network connection? How did you solve the problem? Did you use a protocol analyzer?

Generally, a protocol analyzer is overkill for most network problems. Alternates to protocol analyzers are the relatively new network management systems. IBM's LAN NetView is one of these systems.



Stacey Taylor

In this article, we will develop and debug a management system that uses the LAN NetView network management platform. The remote agents that will be a part of our management system will use the industry standard Simple Network Management Protocol (SNMP).

FOR A FEW DOLLARS MORE (OR LESS)

In a standard distributed network management system, there are a managing workstation and several managed workstations. The managing station requires one of several network management platforms: HP's OpenView, IBM's NetView/6000, IBM's LAN NetView, and SunNet Manager.

In choosing the most appropriate platform, my company, Network Telesis Inc. (NTI), decided to port its LAN/WAN Response Time Monitoring system to the

LAN NetView platform because it is based on an industry standard (XOM, XMP, and X.400), supports OS/2 Presentation Manager, and provides the best user cost/benefit ratio.

Generally available with most TCP/IP products, SNMP agents collect a wide array of information about the workstation they are running on. Over 175 items are in the SNMP information base, including the workstation's owner, the speed of interface card, and the number of bytes received and sent by the card.

But if the remote workstation is on Mars, the information is useless—unless you can extract this information to your local workstation. By using IBM's LAN NetView and the program provided here, you will be able to interrogate any SNMP agent.

The development and operating environment is shown in Figure 1.

The only module of LAN NetView that is used is the **Manage** module. For the remote managed stations, you can use any SNMP-compliant agent running over TCP/IP. An SNMP agent is required to support a predetermined set of data (which is referred to as **MIIB-II** and defined in RFC 1213, available on the Internet).

To send a request to the SNMP

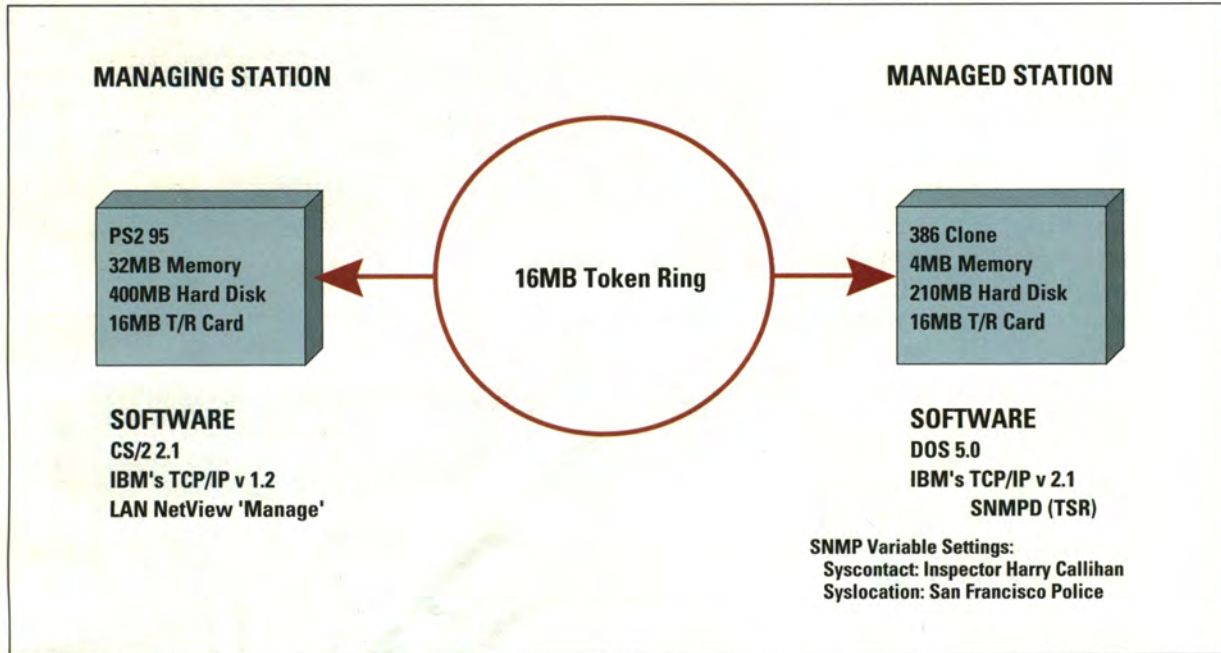


Figure 1. Development environment

agent, LAN NetView requires us to follow a special format when constructing the request. The format is defined by the X/Open XMP specification and is shown in Figure 2.

THE UNFORGIVEN

Our program will read these four MIB-II variables from the SNMP agent: system contact, system name, system location, and number of bytes received by

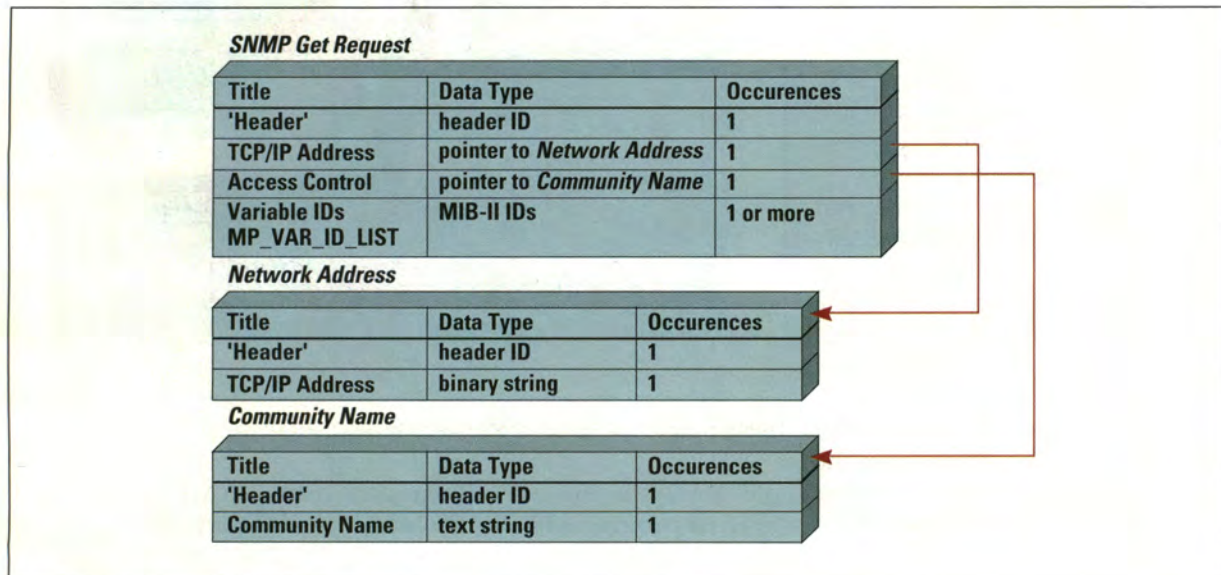


Figure 2. Formatting instructions for SNMP Get



the network interface card.

Once you understand the somewhat obtuse X/Open communications structure, programming within this environment is relatively easy. The difficult part is debugging a problem in the X/Open structure. A timesaving hint in debugging is to code and test the communications modules under OS/2 character mode. Simply starting LAN NetView's GUI (called View) is a very slow process, and it must be terminated before every compile! After the basic communications testing passes in-house testing, the code can be converted to run under the SOM-based View.

Our programming example concentrates on the debugging aspects of SNMP communications. This program contains a malformed SNMP Get request. To assist us in determining where the error is, a routine was developed to convert the SNMP request into a format for easy examination. The routine is called just before the LAN NetView communications instruction and just after the return from LAN NetView. Figure 3 shows the output from our debugging routine.

The LAN NetView error message that appears on the screen points us in the direction of a badly constructed SNMP request. We now need to compare the data in our SNMP Get request against the X/Open structure.

In Figure 3, we can see that there is an obvious error in the seventh line. We coded this as `MP_VAR_BIND_LIST`, which is not a valid entry according to Figure 2. This should have been coded as `MP_VAR_ID_LIST`.

Another error that needs to be corrected is in the eighth line. The MIB-II variable ID has been entered incorrectly. It should be coded 1.3.6.1.2.1.1.6.0.

The SNMP response format, shown in Figure 4, and code sam-

```
1 OM_CLASS OM_S_OBJECT_IDENTIFIER_STRING MP_C_SNMP_GET_ARGUMENT
2 MP_RESPONDER_IP_ADDRESS OM_S_OBJECT 2006C
3 OM_CLASS OM_S_OBJECT_IDENTIFIER_STRING MP_C_NETWORK_ADDRESS
4 MP_IP_ADDRESS OM-S OCTET_STRING 198.202.254.1
5 MP_VAR_ID_LIST OM_S_OBJECT_IDENTIFIER_STRING 1.3.6.1.2.1.1.4.0
6 MP_VAR_ID_LIST OM_S_OBJECT_IDENTIFIER_STRING 1.3.6.1.2.1.2.2.1.10.1
7 Error in determining type! OM_S_OBJECT_IDENTIFIER_STRING 1.3.6.1.2.1.1.5.0
8 MP_VAR_ID_LIST OM_S_OBJECT_IDENTIFIER_STRING 1.3.6.1.2.1.1.60
9 MP_ACCESS_CONTROL OM_S_OBJECT 20090
10 OM_CLASS OM_S_OBJECT_IDENTIFIER_STRING MP_C_COMMUNITY_NAME
11 MP_COMMUNITY OM_S_OCTET_STRING public
```

Figure 3. SNMP error output

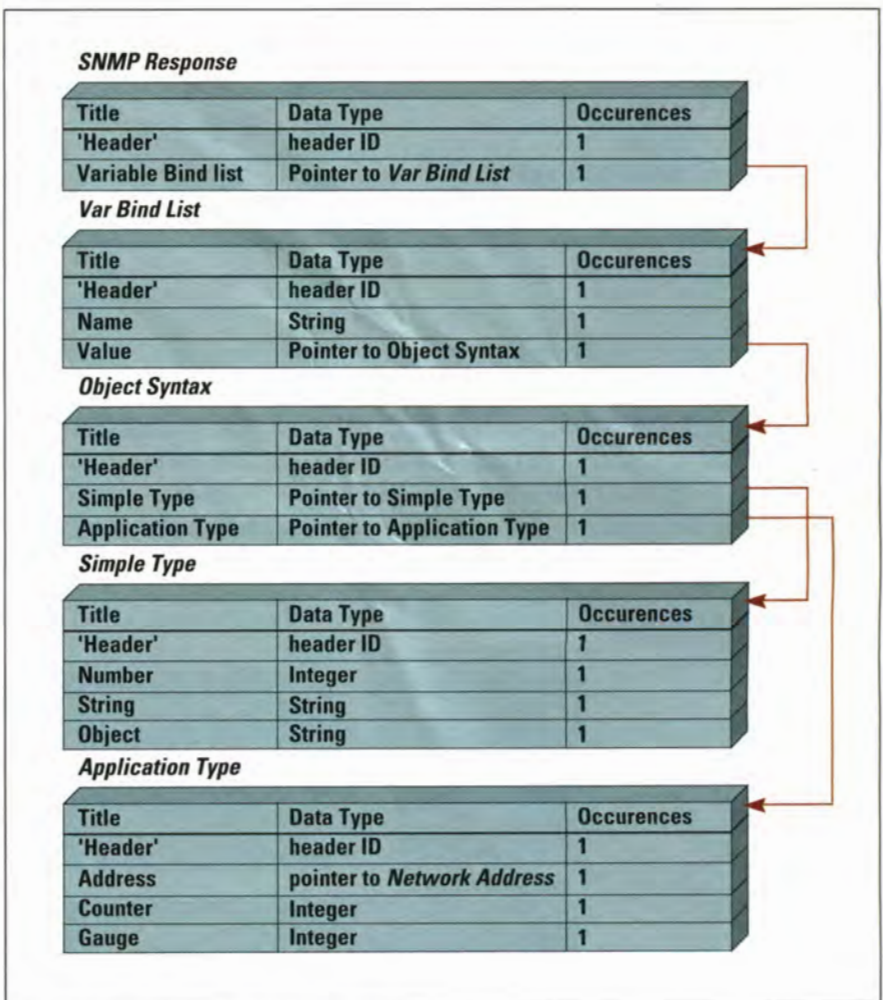


Figure 4. X/Open format for SNMP responses

ple, shown in Figure 5, illustrate how to programatically traverse the SNMP structures. If you grasp the recursive nature of the XMP pointer, extending this routine for

other XMP formats is very straightforward.

The C module that contains the debugging routine has a helper routine that extracts the SNMP

ARE YOU USING THE RIGHT TOOLS



CONFERENCE & EXHIBITION

FIND OUT,

*at the event for professional builders and
integrators of today's software for the enterprise.*

March 14 - 18, 1994
San Jose Convention Center

“The widest variety of course offerings by the greatest minds in the industry. Exposure to some of the best products on the market.”

— Todd Stuart
Sr. Programmer Analyst
UNUM Life Insurance

“A fantastic conference! The ONLY show developers should attend.”

— Brian Francis
Software Engineer
Harbinger EDI Services
Atlanta, Georgia

BUSINESS SOFTWARE SOLUTIONS

*Constructing the Framework for
Client/Server Success*

*Building Distributed Business
Applications*

*Reengineering Business
Information Delivery*

*Leveraging New Technologies for
Competitive Advantage*

*Closing the Gap between
IS and Users*

With a whole new generation of development tools evolving, learning how to choose and use the right tool for the job is more than half the battle. Together, Software Development'94 and Business Software Solutions is the place to go for help. You'll find:

250 vendors representing the spectrum of today's development tools including Microsoft, Borland, IBM, Powersoft, Computer Associates, Apple, Visix, Metaphor, Gupta, Novell and literally hundreds more...

300 practical technical classes on all of today's development challenges ranging from client/server, C++, programming for Windows, managing software projects and teams, object construction and reuse and reengineering IS...

25,000 of your peers attending the software development event of the year.

Keep up with the evolution of software development and integration by attending SD'94/Business Software Solutions. Call or Fax today for more information.

phone 1-800-441-8826
fax 1-415-905-2222

Please refer to this ad when you call or fax.

SOFTWARE DEVELOPMENT

C++

GUI Design

Client/Server and Interoperability

Programming for Windows

Analysis and Design

Mac Technology

OOP

UNIX

OS/2



response data called `GetSNMPData`. Our program shows its use when the returned data is displayed.

As you can see, the SNMP structure is somewhat foreign to most normal structures, but it is easy to use once you understand the basics. And, in most cases, debugging of the communications part is just as easy.

You can now use this fully functional SNMP management program that you can extend to access any MIB-II variable. You can also access any private SNMP extensions that a vendor may have added as long as the variable IDs are provided.

For a good book on SNMP from a programming point of view, I recommend Marshall Rose's *The Simple Book*. If you plan to create any LAN NetView-based management programs, you will need the *LAN NetView Developers Guide*, the XOM and XMP specifications manuals from the X/Open Company Ltd., and the Internet specifications relating to SNMP (such as RFC 1213 and RFC 1157).

Although the listings presented here are only excerpts, the entire program (with `make` file) is available on CompuServe in the OS/2 Developer forum, OS2DF2, in the file `LNVSMP.ZIP`.

THE GOOD, THE BAD, AND THE UGLY

Overall, LAN NetView provides a good entry-level management system. The platform is stable from developer and user viewpoints. IBM does provide applications with LAN NetView, but they are generally limited to local LANs only or OS/2-based WANs.

The current lack of useful LAN NetView applications supporting WAN-based DOS and Windows agents reflects a great deficiency. Only vendors that are providing SNMP-based agents will fill this requirement.

Recently, it has been reported that LAN NetView was to be

```
// This routine will search the SNMP structure looking for a
// match with the desired ID. When found, the ID's data is
// returned in Data parameter.
```

```
int GetSNMPData ( OM_workspace Workspace,
                  OM_public_object SNMPStruct,
                  char *ID, char *Data )
{
    int Found = 0;
    OM_object_identifier OID;

    Data[0] = '\0';

    while ( SNMPStruct->type != OM_NO_MORE_TYPES
            && Found == 0 ) {
        switch ( SNMPStruct->type ) {

            // This can be thought of as a 'Header' for SNMP
            // data that will follow (recursive).
            case MP_VAR_BIND_LIST:
                Found = GetSNMPData ( Workspace,
                                      SNMPStruct->value.object.object,
                                      ID, Data );
                if ( Found == 1 ) {
                    return Found;
                }
                break;

            // If MP_NAME, then this contains the ID of the
            // value returned. If the IDs match, then the next
            // element must be the value, which therefore must
            // have the type MP_VALUE. If this is true, then
            // recurse to get the value. We will then be
            // looking for MP_SIMPLE elements.
            case MP_NAME:
                at_str_to_oid ( ID, &OID );
                if ( at_oid_match ( &OID,
                                   &SNMPStruct->value.string ) ==
                    OM_TRUE ) {
                    SNMPStruct++;
                    if ( SNMPStruct->type == MP_VALUE ) {
                        Found = GetSNMPData ( Workspace,
                                              SNMPStruct->value.object.object,
                                              ID, Data );
                        if ( Found == 1 ) {
                            at_free ( OID.elements );
                            return Found;
                        }
                    }
                }
                at_free ( OID.elements );
        }
    }
}
```

Figure 5. Code extract of SNMP decomposition routine (continued on page 52)

Client/Server Database Solutions

It's available now – ready to perform on your desktop. A new function-rich, 32-bit relational database you can really trust with your growing client/server network, your mission – critical data *and* your business.

Introducing IBM DATABASE 2™ OS/2® (DB2/2™) from IBM Software Solutions, the birthplace of relational database technology.

DB2/2 includes an industrial-strength DB engine that supports transaction management, concurrency control, security, integrity, and recovery functions. Designed to exploit the power and open architecture of OS/2, it also supports industry-standard SQL for developing portable applications. And it runs your DOS, DOS Windows™ and OS/2 applications requiring *online* access.

You can access data directly from DB2/2 on your desktop or from a DB2/2 server on your LAN, and with

DB2

DB2/2
RDBMS

goes

DISTRIBUTED DATABASE
CONNECTION SERVICES/2,™

from DB2®, SQL/DS,™ and OS/400®
databases as if they were on your desktop, too.
This versatility can play a significant role in
an Information Warehouse™ solution
for your business.

We've developed an

desktop.



exciting demo diskette to show you just how well new DB2/2 performs – right on your desktop. Call us today for your free demo, or to order DB2/2: 1-800-342-6672; or fax: 1-800-445-2426. In Canada, call 1-800-465-7999, ext. 850. An upgrade from OS/2 Extended Edition or Extended Services is also available.



replaced by an OS/2 version of IBM's AIX-based NetView/6000. The truth is that the SNMP APIs in NetView/6000 will be ported to LAN NetView. This addition to LAN NetView should strengthen its SNMP capability.

You now have the basic abilities to query any SNMP agent on your network. Your next task is to display the results in LAN NetView's View display. The *View User Interface Developer's Guide* shows you how.

Stacey Taylor, *Network Telesis Inc.*, Calabasas, Calif., is a senior software engineer for the development of the company's *Response Time Analysis* product for LAN NetView. He can be reached on CompuServe at 73670,3235.

REFERENCES

"Introduction to LAN NetView Application Development: An XMP/XOM Primer," Jacob M. Rzepka and Chuck R. McKelley, *OS/2 Developer* magazine, Vol. 5, No. 5, November/December 1993.

Marshall T. Rose, *The Simple Book*, Prentice Hall, 1994.

LAN NetView Developers Guide (IBM Document S96F-8493-00)

X/Open OSI-Abstract-Data Manipulation API (XOM) (XOpen Document C180).

X/Open Systems Management Management Protocols API (XMP) (XOpen Document P170).

```
        break;

        // 'Header' that means data following is of simple
        // or application type
        case MP_APPLICATION_WIDE:
        case MP_SIMPLE:
            Found = GetSNMPData ( Workspace,
                                SNMPStruct->value.object.object,
                                ID, Data );
            if ( Found == 1 ) {
                return Found;
            }
            break;

        // This is the data !!
        case MP_STRING:
        case MP_COUNTER:
        case MP_NUMBER:
        case MP_GAUGE:
            Found = 1;
            break;

    } // type switch

    if ( Found == 0 ) {
        SNMPStruct++;
    }

} // while loop

if ( Found == 1 ) {
    ExtractValue ( Workspace, SNMPStruct, Data );
}

return Found;
}
```

Figure 5. Code extract of SNMP decomposition routine (continued from page 50)

READY NOW! From the publishers of

OS/2 Developer/Software Development/LAN Magazine/DBMS/

Database Programming & Design/ Dr. Dobb's Journal/UNIX Review and IBM Corp.!

Now your search for OS/2 solutions just got easier.

Now you can access
the most comprehensive,
detailed listings of vendors
and applications serving
the OS/2 market — all from
one source: the IBM OS/2
APPLICATIONS DIRECTORY.

This is the complete guide to solutions for even the most specialized applications. It keeps developers up to date on the competition and end-users up to speed on what's available and where to find it.

The 270-page directory gives you fully detailed listings that include:

- Company name
- Contact name
- Address
- Phone number
- Product description
- Details on software features
- Price
- Availability date



Applications are conveniently organized by 17 general industry categories in alphabetical order:

- Accounting/personal finance
- Business-specific
- CAD/CADAM
- Communications

Also Available
On CD-ROM — includes
over 1,000 actual listings
with LAN 3.0 certified products.
Complete electronic brochure for each product.
Made by Selectware™

- Database
- Development tools
- Electronic mail
- Games/entertainment
- Graphics / Desktop Publishing
- Information management
- Integrated workgroup
- Networks
- Spreadsheets
- Utilities
- Word processing

Plus you get:

- Index of vendors, cross-referenced by product name
- Special separate section of applications certified by IBM's Ready! for LAN Server certification program

Make your solutions-search easy.

Send for the IBM OS/2 APPLICATIONS DIRECTORY.

OS/2 is a registered trademark of IBM Corp.

ONLY \$9.95 FOR BOOK, \$19.95 FOR CD-ROM

☐ Book \$9.95 ☐ CD-ROM \$19.95 ☐ Both \$29.95

Please indicate number of books or CDs desired.

US\$3.50 shipping/handling. Add US\$2 for shipping to Canada and overseas.

☐ Payment enclosed

(include sales tax in CA, GA, IL, NY, & TX; GST in Canada)

☐ Charge my: ☐ Visa ☐ MasterCard ☐ American Express

Card #

Exp. date

Name

Title

Company

Address

City/State/Prov.

Zip/Postal Code

Country

Phone

FAX

Signature (REQUIRED)

MAIL TO: Daniel Strickland, OS/2 Applications Directory, Directory Dept.,
Miller Freeman Inc., 600 Harrison Street, San Francisco, CA 94107 USA
FAX TO: (415) 905-2239 OR CALL: (415) 905-2728

GAME DEVELOPER

TAKE YOUR TURN IN THE **\$10 BILLION+** COMPUTER GAME INDUSTRY!



Introducing . . . Game Developer, the magazine for programmers and developers of console, arcade, and home computer games.

- Profile of ID software
- Designing a multi-player game
- The business of game distribution
- Sound chip programming
- The making of Iron Helix

This publication is the information source for game programmers and developers looking for the latest trends in code, commerce and creativity.

ORDER YOUR COPY TODAY!

YES! Send _____ copies at \$6.95 each = \$ _____
(Foreign add \$3.00 each) \$ _____
TOTAL DUE \$ _____

Name _____

Company _____

Address _____

City/State/Zip _____

Prepayment Required

- | | |
|---|-------------------------------------|
| ☐ Check enclosed | ☐ Charge my: |
| ☐ VISA ☐ MC | ☐ AMEX |

Card # _____ Exp. _____

Signature _____

Please make checks payable to Miller Freeman, Inc. Orders must be prepaid in U.S. Dollars. Please allow 3-4 weeks for delivery. Price includes domestic shipping & handling.

To order: Telephone 1-800-444-4881 -- or -- Fax 1-415-905-2233 -- or -- Mail this order form and payment to: Game Developer, P.O. Box 105448, Atlanta, GA 30348-5448



This article contains practical information for network application developers, including coding errors to avoid and answers to typical configuration management problems. Developers will also find the tracing information and examples of classic errors helpful when debugging their network applications.

By TOM ELLIS and MIKE MCINTYRE

NetBIOS and 802.2 Applications: Problems and Solutions

If you are a developer of network software that uses either NetBIOS or 802.2 APIs, you may want to consider the following tools and advice. For those of you who are writing your first network application, we include some samples and information that you may find useful. In addition, we dug into our files to present common and unusual problems encountered by programmers and offer some potential solutions.

If you are just getting started on a network application, there are some basic sample programs available for NetBIOS and 802.2 that can save you a lot of time. Although they come with the *IBM LAN Technical Reference*, they have been updated to be either 16- or 32-bit and now are provided with *makefiles*, *LRF* and *DEF* files, and detailed instructions.

There are also several *INC* and *H* include and header files that contain all the data structures for C and MASM source. IBM will reissue the sample programs and include files with the updated *LAN Technical Reference*. The programs and files are available now on CompuServe in the *OS/2 Developer* forum, *OS2DF2*. The *OS2DF2* section also contains some of the latest fixes of transport modules as well as descriptions for these files.

SYSTEM TRACING FACILITY AND SNIFFERS

Many programmers are not aware of the system tracing facility that is part of the OS/2 base operating system. Different from the Communications Manager traces, the system tracing facility helps track commands and frames visible at the transport layer and records different types of system events. The major event code for NetBIOS and 802.2 events is 164, which is activated by typing *TRACE ON 164* at an OS/2 command prompt.

So, what can the 164 traces do for you? This tracing shows network activity with NetBIOS and 802.2 *ring 3* APIs and the link control layer. That is, it shows what happens to your commands and whether they result in any network-frame activity. There are a variety of network sniffers in use, but they can only track the actual frames on the wire without connecting them with a specific application.

Sniffers are helpful, if you suspect that frames are not making it over a series of bridges or if you want to monitor the link-level activity closely for some reason. They have a relatively large storage capacity as well as several filtering capabilities. The 164 traces have a smaller buffer but give you an exact picture of interactions with your transport applications. The 164 traces trace API commands, frame headers, and user



data up to 128 bytes, whereas network sniffers trace frames only and cannot trace the transport APIs. These APIs can be command control blocks (CCBs) for 802.2 or network control blocks (NCBs) for NetBIOS. When you are developing a new network application, 164 tracing is the tool to use.

To setup and collect a transport trace, you need to follow certain steps. First, include TRACEBUF=63 in CONFIG.SYS to allocate space for system trace buffer. Second, include in PROTOCOL.INI under the LANDD or NETBEUI section header the appropriate trace keyword and trace mask. The *NTS/2 LAN Adapter Support (LAPS) Configuration Guide* explains these keywords and masks. Third, type TRACE ON 164 at an OS/2 command prompt to start tracing. You can automatically start tracing at system IPL time, if you include TRACE=ON 164 statement in the CONFIG.SYS file.

Once the event you wish to trace has occurred, stop the trace by typing TRACE OFF at an OS/2 command prompt. Next, type TRACEFMT. This invokes a Presentation Manager program that allows you to save your trace data to a disk file in either flat ASCII or binary form. If you plan to format the data further by using DTF3.EXE, save the trace data to an ASCII file before using DTF3.EXE. Then, invoke DTF3.EXE to produce a formatted trace data file called TRACEFMT.DTF from an input trace file called TRACEFMT.SAV by typing DTF3 TRACEFMT.SAV > TRACEFMT.DTF at an OS/2 command prompt. You now have a trace that shows CCB/NCB level detail at the transport APIs.

Since the trace buffer wraps within seconds, we've added some parameters to make 164 traces easier to use. One is TRACEOFF, which is valid for 802.2 and NetBIOS. TRACEOFF turns off traces when a specified return code is detected. The other is TRACENAMES, which is specifi-

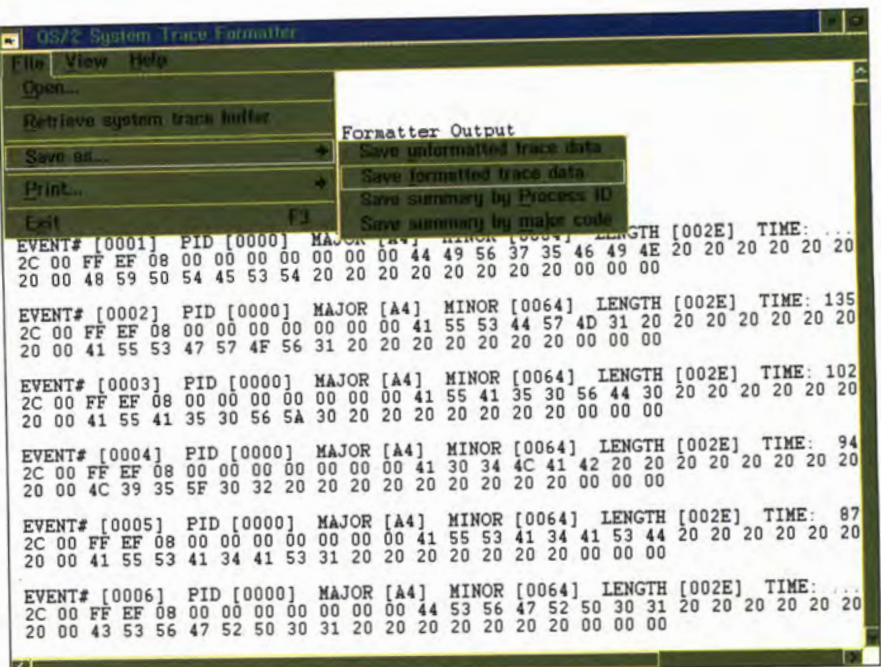


Figure 1. TRACEFMT output

cally for use with NetBIOS. TRACENAMES filters tracing to activity between the local machine and a specified remote NetBIOS name, thus eliminating irrelevant frames from the trace data and helping to save buffer space. You set these parameters by inserting them into the appropriate place in PROTOCOL.INI. See the *LAN Technical Reference* or the README file in the NTS/2 corrective service diskette (CSD) for details.

Figures 1 and 2 show two examples of trace output. Figure 1 is an example of the output that the TRACEFMT program gives after a 164 trace has been taken. Figure 2 is an example of the output from the DTF3.EXE program.

PROBLEM DETERMINATION

At this point, you're testing your application on a network that contains several, if not hundreds, of nodes connected with series of bridges. Something goes wrong. How do you locate the problem?

When isolated, the problem will fall into one of six categories. The first four types of problems are

solvable when you complete the diagnosis. The remaining two require help from IBM support.

Configuration and Resource Management. After noticing a problem that may be transport-related, look in the LANTRAN.LOG file for possible messages that may direct you to adjust some configuration parameters in PROTOCOL.INI.

If the problem is not apparent in LANTRAN.LOG, try reproducing the problem with the 164 traces active. Once formatted, the trace may show a bad return code or notification of a status event that you can look up in the *LAN Technical Reference*.

Application Program Problems. If your application traps, causing a pop-up to appear on the screen, but the system is still running, it's very likely that you have an application bug. The CSLIM gives the offset in your program where the trap occurred. If you get a trap that stops the whole system, an inadequate stack size passed in by the application could be the cause. Pointers passed in by the application could trap the machine, if they



World's Largest DB2 Family Conference

IDUG

DB2—Serving the Changing Enterprise

6th Annual

North American Conference

May 8 -12, 1994

San Diego Marriott

& Hyatt Regency

Hear More than 100 Unique User Experience Presentations

- DB2 Version 3
- Client/Server
- DB2/2 & DB2/6000
- Connectivity
- Distributed Heterogeneous Systems
- Performance and Tuning
- Database/Application Design
- Database Administration

Don't Miss These Enlightening Keynote Presentations

- "The DB2 Family on the Move — Tools for the Impatient"
Presented by Steve Mills, IBM Software Solutions Division
- "Computer Downsizing Architectures and Technologies"
Presented by Larry DeBoever, DeBoever Architectures
- "DB2 Performance Tuning Suggestions" or "What If Your
DB2 Task Keeps Going and Going and Going?"
Presented by Stu Hoyt, Independent Consultant

Educational Seminars

- An Overview of DB2 Version 3
- DB2 Internals for DBAs
- Advanced SQL
- Achieving Continuous Availability (24x7)
- DB2 Application Design for Performance
- Gateways to DB2—DDCS/2 and DDCS/6000
- Introduction to DB2/2 and DB2/6000

Talk with the IBM Developers

Visit the Vendor Exhibits

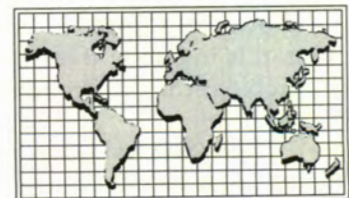
Discuss Issues with Industry Experts

Submit Your Questions for Panels & SIGs

Call 312-644-6610

or 312-527-6777 (voicemail)

to register or receive a conference brochure.



**INTERNATIONAL
DB2 USERS GROUP**

An Independent, Not-for-Profit, User-Run Organization



are invalid. You really need to find out what kind of trap has occurred to determine whether it is the application or transports.

Hardware Problems. Yes, many times an erratic problem plagues a certain machine, and it turns out to be a bad adapter or cable. Bad adapters can run for several hours without error and then suddenly deliver corrupt data. Bad cabling can cause intermittent beaconing or lobe wire faults on Token Ring; on Ethernet, it can prohibit connectivity between certain machines. Make sure you have those wretched terminators attached.

Bridge Malfunction/Configuration. To diagnose bridging problems, you really need sniffers or similar network analyzers on both sides of the bridge. In this case, the one node sends frames, but the frames never arrive at the other side. Determine in which segment in the network the frames fail to appear. Then, look at the bridge between those two segments to isolate the problem to the particular bridge.

Traps and System Lockups. These types of traps stop the entire system. If your system traps and displays trap-register information, you may be able to tell in which specific driver the trap occurred by looking at the CSLIM value. You need to use a debugger, such as the Kernel Debugger, to debug traps and system lockups. IBM will also be able to aid in problem determination for traps and system lockups. To determine if the problem may be in the transports, check the CSLIM against the values shown in Figure 3. Even if the trap is in one of these drivers, the problem can still be related to the application.

The CSLIM values in Figure 3 are for LAPS version 2.11 (NTS/2). NETBEUI and LANDD have values for 16-bit and 32-bit code selectors.

If you have a release of LAPS later than 2.11 these values may be slightly larger. Media access control (MAC) drivers can also cause system traps infrequently, but you will need to call support to make that determination. None of these drivers traps in normal operation.

Base Operating System. If the CSLIM value is not close to one of the values shown in Figure 3, it is possible that the trap is in the base operating system. Don't count on this, since the base system rarely traps because of an application error. Still, if your machine is completely locked up, support can help you determine if the base system is involved.

EXAMPLES OF CLASSIC NETBIOS ERRORS

Reusing NCB before it is complete.

A common NetBIOS application bug is reusing an NCB before it is complete. When the NCB does complete, it looks like the second one completed. This can cause any number of undesirable events that you may not be aware of. It might look like you did

everything correctly and that NetBIOS just corrupted your data. This can happen in a number of ways. One is queuing the completed NCB somewhere and assuming it will get processed before the new one with the same address gets written to.

On no wait command, check completion code, not return code. Your no-wait NCB never completes, but you received the data.

Although it sounds simple, make sure you checked the completion code and not the return code.

The ACSNETB.DLL is there, but the NetBIOS device driver (DD) is not. You keep issuing commands to ACSNETB.DLL, and some of them complete, but all the others give catastrophic return codes. And you know this application works on other machines.

Maybe the NetBIOS device driver or a MAC driver did not load. Some commands will return immediately when issued to the dynamic link library (DLL), so it looks like some commands are correct. Check the LANTRAN.LOG file to



Figure 2. DTF3 output

Filename	16-bit	32-bit
LANDLDD.OS2	6a73	
NETBIOS.OS2	21b0	
NETWKSTA.200	1e85	
NETBEUI.OS2	916c	b229
LANDD.OS2	b5b8	d047

Figure 3. CSLIM values for LAPS version 2.11

see if the drivers loaded.

Setting priority thread too high in application program. Your application hangs, waiting for NetBIOS to complete some command. Did you set your thread priority higher than NetBIOS itself?

There are two different priorities set for EE 1.3 and LS 2.0 and above. The LS 2.0 and later priority is lower than EE 1.3 NetBIOS, so be aware of this when moving an application to the NDIS version and setting thread priorities.

No outstanding Receive.any. If your NetBIOS program will not accept data or is running unusually slow, a Receive command that was not issued quickly enough may be the cause.

When NetBIOS data arrives at your machine, NetBIOS will accept it only if you issue a Receive command with sufficient buffer space available to hold the data. It is good practice to reissue a Receive on the appendage when a Receive completes, so you will be less likely to miss incoming data.

Not enough loopback packets configured. You are using a redirected drive that is local to the LAN Server machine, but the application just hangs, and you can't get access. When you try to get to that drive from a remote machine, everything is fine.

This looks like the NETBEUI loopback packet is set at the default setting of 1, which should be raised to 5, for example, if you are actually using the loopback feature.

Message number LT00065 shows up in LANTRAN.LOG. Usually this means that the resources don't fit into the NetBIOS 64K data area. This is not the same as the NETBEUI RAM calculations for NetBIOS, as given in the NTS/2 configuration guide. There is only one 64K data area for all adapters for the NETBIOS.OS2 module.

To determine the resources to allocate for NetBIOS, the sum of the following calculations must be less than 65,536:

- 11200 bytes overhead
- 95 bytes for each NCB available to NetBIOS applications
- 15 times (NCBs available minus 15). Use 0 if less than 0
- 990 for each adapter configured for NetBIOS.

EXAMPLES OF CLASSIC 802.2 ERRORS

Direct Station use in 802.2 (DOS vs. OS/2). When programming to the 802.2 API in DOS Direct Station, you obtain ownership by issuing DIR.OPEN.ADAPTER. In OS/2 Direct Station, you obtain ownership by explicitly issuing DIR.OPEN.DIRECT. Also, keep in mind that only one application can own the Direct Station on an adapter at one time. This is important in determining which applications can coexist on one adapter.

Freeing service access point (SAP) buffers twice. A common problem arises when using buffers from the SAP buffer pool. When users need a buffer to transmit data, they use BUFFER.GET to transfer ownership of the buffer from the protocol stack to the user. Once the

Footprints



for OS/2

- Tired of **slow** debuggers?
- Can't duplicate a reported customer problem?
- Need to verify all paths of execution without using *printf*?
- Trying to support code in Dime Box, Texas, from Brooklyn, New York?

Footprints is the product for you!

Dynamic debugging
Online monitoring/Offline analysis
Realtime systems
Telecommunications
Multi-thread / multi-process
Remote support and postmortems

Dynamically select which portions of code to trace - in a live user environment!

Footprints is simple to use! Just include a header. Link to a DLL. You're ready to call **Footprints** and begin capturing any or all the data you require. **Footprints** doesn't bog down your code with constant tracing or file I/O. The tracing only gets activated *when and where* you want it. From our friendly **PM interface**, you dynamically turn on one or more of up to 64 separate trace ids that control what data you are tracing. You can then view the time-stamped data in memory as it whizzes by, or view it at your leisure from a file days after the fact.

ONLY \$ 99

Call **713-871-1448** now!

VISA/MC or send a check to
Hardy Software Systems, Inc.
4801 Woodway, #415 • Houston, TX 77056
s/h charge \$ 5.95
Total Cost \$104.95
Texas residents add 8.25% sales tax.
Requires OS/2 2.0 or later.



Hardy Software Systems, Inc.
Circle Reader Service Number 26

Transitioning to Smalltalk technology?
Introducing Smalltalk to your organization?
Travel with the team that knows the way...

The Object People

"Your Smalltalk Experts"

SMALLTALK

Education & Training

- Smalltalk/V Windows and PM
- PARTS
- Objectworks\Smalltalk
- VisualWorks
- Smalltalk for Cobol Programmers
- Analysis & Design
- Project Management
- In-House & Open Courses

Project Related Services

- Consulting & Mentoring
- Rapid Prototyping
- Custom Software Development
 - Legacy Systems
 - GUI's, Databases
 - Client-Server

The Object People Inc. 509-885 Meadowlands Dr., Ottawa, Ontario, K2C 3N2
Telephone: (613) 225-8812 FAX: (613) 225-5943

Smalltalk/V and PARTS are registered trademarks of Digitalk, Inc.
Objectworks and VisualWorks are trademarks of ParcPlace Systems Inc.

Circle Reader Service Number 27

Looking for
fast answers
to your
development
questions?



**Golden
CommPass**
OS/2 Navigation Software

You need to do two things. Go on-line with CompuServe Information Service. Use Golden CommPass to do it.

CompuServe hosts OS/2 developer and user forums, monitored by technical personnel. IBM developers are there with responses to anything that's got you stuck. Other OS/2 programmers and enthusiasts are there, too, comparing notes and giving feedback. It's definitely the place to be.

Time is money when you connect to CompuServe, and Golden CommPass saves you both. It lets you compose your questions off-line, send them in a flash and disconnect. It gets your replies while you're busy programming.

Golden CommPass keeps your development schedule on course. The fact is, the sooner you start using our program, the sooner they'll be talking about yours.



Creative Systems Programming Corporation
Post Office Box 961 • Mount Laurel, NJ 08054-0961
(609) 234-1500 • Fax: (609) 234-1920 • CompuServe: 71511,151



Circle Reader Service Number 28

users are finished with the buffer, they use `BUFFER.FREE` to return ownership of the buffer's memory back to the protocol stack. Also, the protocol stack takes buffers from the pool and fills them with data when inbound data is received. 802.2 does no checking to prevent a user from freeing a buffer more than once. Unpredictable results occur. When a buffer is on the free queue, freeing it again causes 802.2 to create a circular linked list of queue elements and from there many problems crop up.

Reuse of CCB before it's complete. 802.2 will reuse CCB memory before the CCB is completed, which is a tricky application bug. A common symptom of this is that 802.2 appears never to complete a specific CCB. Just as likely, the symptom shows up as a system trap or as unexpected results on command completion.

Applications using same source SAP number. If an 802.2 application opens a specific SAP and another application attempts to open the same SAP on the same adapter, you get a return code of `x'43'`, informing you that the SAP number is already in use.

Completion flag set in CCB may cause out of queue elements error. If an 802.2 application issues CCBs with the command completion field set to a nonzero value, the application wants to be notified of command completions retrieved using `READ` commands. If the command completion field is set and no `READ` commands are issued for command completion events, queue elements are internally accumulated, and eventually a return code is given back stating "out of queue elements".

GENERAL PROBLEMS

Here are some examples of general problems users have experienced on their networks and how they were solved.



Sessions going down on slow links. Several networks had problems with links and sessions going down when data was sent across transparent bridges or slow 64K links. Tracing showed the T1 timer expiring before the link was taken down. The data and the corresponding acknowledgment frame did not get across the line before the retries were exhausted. The solution was to increase the T1 timer. The DLC retries parameter could also be increased, but then there would be lots of unnecessary resends of frames getting sent and discarded.

Deallocating application memory causes a trap. One user caused a ring 0 trap in the NetBIOS device driver. In this case, the program allocated ring 3 memory and issued a command in which NetBIOS would use the memory to complete the command. However, the application freed the memory before the command completed. NetBIOS validated the address on entry, but it was invalid upon completion.

Application does not have response for all possible return codes. An example of a common application bug is inadequate return code handling. Many applications have processing for only a limited number of return codes, but when something unexpected happens, one of these other return codes can be hit. Anything can happen at this

point. Make sure return code handling is detailed enough.

Application fails after going into debugger. If you are debugging a ring 0 network application, and you have problems getting frames to be sent after a command is issued, it might be that the adapter is closed because it timed out. This happens when stepping through code in ring 0. People have wasted hours thinking their application was doing something wrong.

You are running Ethernet DIX but configured for 802.3. You have configured a new Ethernet machine, but it can't communicate with other machines or the frames can't make it across bridges.

It could be that all the other machines and bridges are using the DIX frame format, but you have inadvertently configured the new machine for 802.3. People have spent days figuring out this trivial problem.

LAN Server locks up. You are running a LAN Server or Requester machine. Suddenly, it locks up, and you have to power it off. It then runs for a while but soon locks up again. You probably are out of disk space for the swapper or have insufficient RAM. Servers with 6MB of RAM are vulnerable to this.

Bridging and duplicate address errors. You have a large network with many bridges. The machines running 802.2 applications receive

duplicate address errors on boot up. You are certain that you don't have a duplicate address on the network.

Several customers have discovered that a malfunctioning bridge can echo frames generated by the duplicate address test done by LANDD at boot-up. This makes it look to LANDD as if there is another node responding to the address test frames. A sniffer can locate the bad bridge by showing routing information.

LAPS won't install due to locked files. When trying to install LAPS over an existing version of LAPS, the system gives an error message about locked files and ends the install. The message persists even after rebooting.

The solution is to remove the IBMLANLK.LST file from the \OS2\INSTALL subdirectory and reinstall LAPS.

If you've read all this and still think that the transport has a problem, licensed users of NTS/2 and LAN Server can contact IBM Support at 1-800-992-4277.

Mike McIntyre joined the LAN Transport development team at IBM Austin in 1989. He was a developer on several projects including the IEEE 802.2 protocol stack and IEEE 802.2 Virtual Device Driver. He is currently a member of the team working on Multi-Protocol Transport Services for OS/2. He received a B.S. in electrical engineering from Penn State University.

Tom Ellis joined the LAN Transport development team at IBM Austin in 1988. He was the level 3 support lead for the NetBIOS and 802.2 interfaces during the last two years. He is currently working as a developer on enhancements to the NetBIOS interface. He graduated from Ohio State University in 1988 with a B.S. in computer and information science.

REFERENCES

IBM LAN Technical Reference, SC30-3587-00.

NTS/2 LAN Adapter and Protocol Support Configuration Guide, S96F-8489-00.

IBM Token-Ring Architecture Reference, SC30-3374-02.

OS/2 PRODUCTS & SERVICES

IRMWARE SERVICES

EDPT - Employee Development Planning and Tracking a low cost solution to fully automate Human Resource procedures (may be customized to meet your needs).

E/R DIAGRAMMER - An inexpensive entry into CASE technology that remains methodology independent without losing the precision of binary cardinalities.

IRMWare specializes in custom business solutions and software development.

FREE PRODUCT BROCHURE OR CONSULTING INFORMATION:
(602) 730-1510 or Fax (602) 413-0399 • P.O. Box 51093 • Phoenix, AZ 85076-1093

Circle Reader Service Number 32

John C. Dvorak's Favorite

"The best of the best...Hobbes OS/2 CDROM... Lots of nifty utilities, fonts, and other goodies on this CD." -John Dvorak, PC Computing, Oct. 93.

Hobbes is 2800 files from the largest ftp archive of OS/2 material, ftp-os2.cdrom.com, quaintly known as Hobbes. Hobbes serves as the major internet distribution site of OS/2 freeware, shareware, device drivers, program updates and information. We bring this entire archive to you in the well organized, indexed, and virus scanned Hobbes disc. Many programs are installed on the CDROM, so you can run them without using any of your hard disk space.

The Hobbes OS/2 CDROM is updated quarterly and last updated in October 1993. Start enjoying this disc now. We guarantee your satisfaction.

\$29.95 Many more products available.

Walnut Creek CDROM • 4041 Pike Lane, Ste D-98 • Concord, CA 94520
1 800 786 9907
+1 510 674 0783 FAX: +1 510 674 0821 info@cdrom.com

Circle Reader Service Number 31

MISSING SOMETHING?

Complete back issue library available today!
Every back issue of OS/2 DEVELOPER can be yours.

Call today and complete your library:
1-800-WANT-OS2

YOUR SINGLE SOURCE FOR OS/2 APPLICATIONS

UTILITIES	PRODUCTIVITY SOFTWARE
BOOKS & VIDEOS	GRAPHICS & CAD
OS/2 T-SHIRTS	COMMUNICATIONS
BACKUP SOFTWARE	NETWORKING
DEVELOPMENT TOOLS	WORD PROCESSING

ORDERS: 1-800-776-8284

FAX: 919-878-7479

INQUIRIES: 919-878-9700

3209 Gresham Lake Road, Suite 315 Raleigh, NC 27615 USA



Circle Reader Service Number 29

Schedule Programs and Reminders ...



Chron v4.0

Scheduling Software for OS/2

New version.....only \$99



Hilbert Computing
1022 N. Cooper
Olathe, KS 66061

Voice: (913) 780-5051
BBS/Fax: (913) 829-2450
CIS: 73457,365

Circle Reader Service Number 30

Attention Mathematica users:
Everything you need to know to get the most out of your math programming is in

THE MATHEMATICA JOURNAL

Call or fax today for information about subscriptions and special issues:
Phone orders: 415-905-2332
Fax orders: 415-905-2233

SUBSCRIBE TODAY!

Only \$39.95 for a full year subscription to the premier source of tools and techniques for the OS/2 software developer: **OS/2 DEVELOPER**

Call today and don't miss a single issue:

1-800-WANT-OS2



REMOTE LAN. NETWORKING

RemoteVision extends LANs transparently over dial-up connections. RemoteVision provides access to a LAN from remote machines or remote LANs. Remote DOS, Windows, and OS/2 machines have a simulated network adapter.

Communications to the LAN is provided by the RemoteVision Server and async modems. Most LAN protocols and applications work, including NetBIOS, Named Pipes, APPC, 802.2, TCP/IP, and IPX. Call (415) 965-8607 and use option 2 for a free fax-back.

Circle Reader Service Number 60

OS/2 PRODUCTS & SERVICES

Opt-Tech Sort/Merge

High Performance Sort/Merge/Select Utility

Use as a stand-alone utility or call as a Subroutine. Many features including unlimited file size, multiple keys, record selection, duplicate elimination, summing and more!

Call for free brochure Opt-TechData Processing
P.O. Box 678
Available for: Zephyr Cove, NV 89448
OS/2 or Unix \$249 Phone (702) 588-3737
DOS or Windows \$149 Fax (702) 588-7576

Circle Reader Service Number 34

The OS/2 Developer Product & Services Guide

The most cost effective way to reach
36,000+ qualified professional
OS/2 Developers

New England/Midwest
call Trish Sutter
(212) 626-2498

Middle Atlantic/Southeast
call Dave Moreau
(212) 626-2318

West/Southwest
call Angela Barnett
(415) 905-4983

FAX for OS/2

From the developers of **FaxWorks OS/2** and **PMfax**

Client/Server API and Printer Driver Toolkits
Support for LANs, up to 48 lines per CPU,
and all popular fax hardware.

Keller Group Inc.

Voice: (612) 429-7273

Fax: (612) 653-1987

FaxWorks is a trademark of SofNet, Inc.

PMfax is a trademark of Keller Group

Circle Reader Service Number 39

GET AN EDGE ON YOUR MARKET

With customized consulting
services for OS/2 & client
server applications:

- Programming
- Project Management
- System Analysis
- Maintenance
- Design
- Testing
- Installation

**REAL-TIME
TECHNOLOGIES**

(800) 441-9241
(708) 441-9240

1001 Greenbay Rd, Winnetka, IL 60093

Circle Reader Service Number 35

TCP/2™

TCP/IP for OS/2®

Supports **all** released versions of OS/2
including 32bit API for OS/2 2.0 and above

Supports NDIS, CM, and ODI drivers
Network access from protected, real, and WINOS2 sessions
Full server operation for **telnet, ftp, rsh** and **sendmail**
IP gateway capability

**NETBIOS, NFS ACCESS, SNMP TOOLS
DOMAIN NAME SERVER**

**DEVELOPERS KIT with socket library
VT102 & 3270** emulation for remote login

TCP™-OS/2 TRANSPORT MODULE.....\$ 200.00

A DAN LANCIANI PRODUCT

For further information contact: Essex Systems, Inc.
One Central Street, Middleton, MA 01949 • (508) 750-6200

TCP/2 is a trademark of DLD consulting. Ethernet is a trademark of Xerox Corporation,
OS/2 is a registered trademark of International Business Machines Corporation. VT102 is
registered trademark of Digital Equipment Corporation. TCP/2 is based in part on
work done by the University of California Berkeley.

Circle Reader Service Number 37

The easy to use 32 bit OS/2 PM IPF Language Editor...

IPF Editor

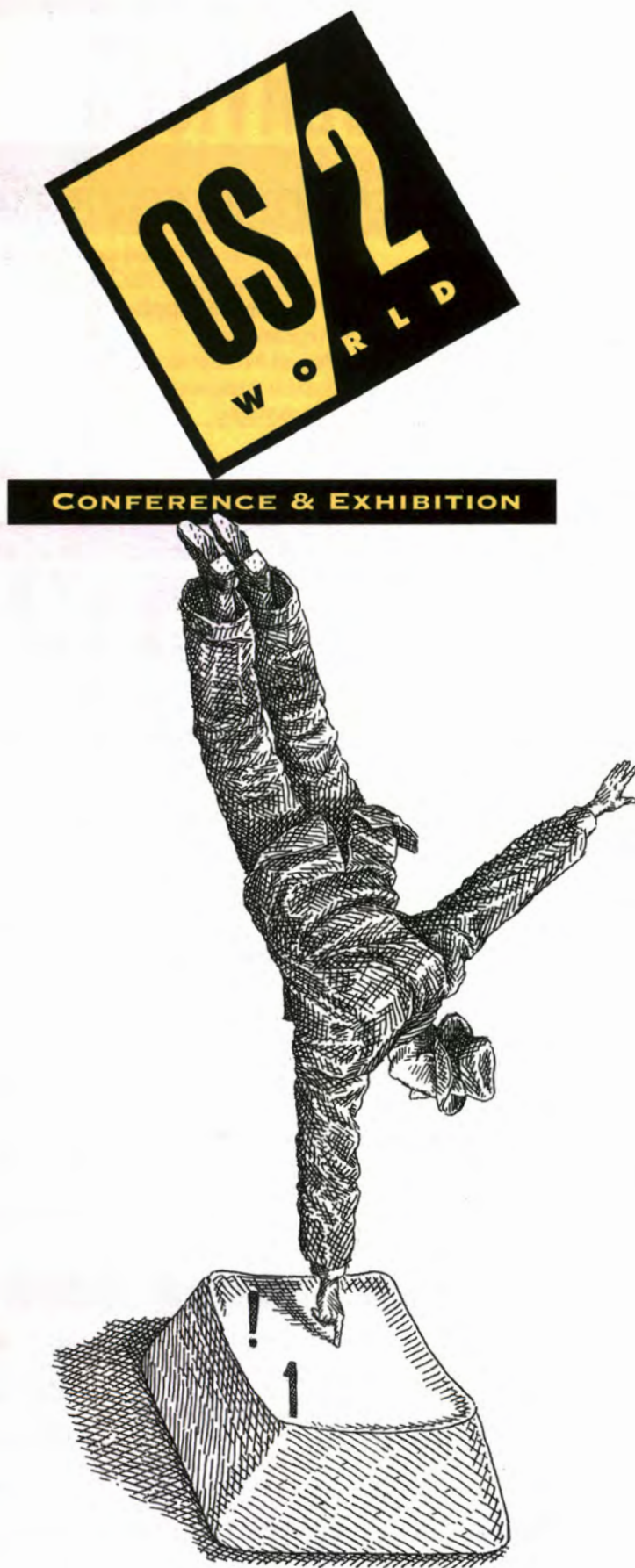
Introductory Price: \$95
With Printed Manual: \$125

- Add help to applications in minutes
- Designed for easy use by programmers and non-programmers
- Generate C and Resource source files to add help to applications automatically
- Import wordprocessing files, including WordPerfect, quickly and accurately
- Preview IPF output without compiling
- Create all hypertext links automatically
- Optional voice and animation modules
- Includes complete on-line help and documentation

**Orders and
Information**
(206) 428-5025
on compuserve at 70410,2416

Perez Computing Services
4725 Monte Vista Place
Mt Vernon, WA 98273

Circle Reader Service Number 61



Announcing

the 1st Annual

OS/2 World
Conference &
Exhibition

July 19-22, 1994

Santa Clara
Convention Center
Santa Clara,
California

Do you want more support in your day-to-day use of OS/2? Is your company struggling to integrate OS/2 into its enterprise computing strategy? Do you want to take OS/2 to a higher level?

OS/2 WORLD WILL POINT YOU IN THE RIGHT DIRECTION

The independent technical program, featuring over 100 lectures, workshops and tutorials, will present you with the objective truth — both good and bad — about OS/2, IBM, and third-party providers. You can count on the information because our top-notch faculty won't have a hidden agenda. More importantly, the technical program is geared to you, the OS/2 customer, not the industry.

The technical program is broken out into 8 dynamic tracks:

- Enterprise-Wide Networking
- Client/Server and Information Management
- Local Area Networking
- Power Computing
- Systems Administration
- Corporate Software Development
- Personal Programming
- Multimedia

AN OBJECTIVE AND BALANCED FACULTY

Classes will be taught almost exclusively by independent expert users, consultants, and trainers offering objective and balanced information to users and managers charged with deploying and maintaining OS/2 systems in corporate environments. Instructors will be experts who work with OS/2 in the field, and have implemented OS/2 solutions for their companies or clients.

INDUSTRY-WIDE SPONSORSHIP GUARANTEES A TOP QUALITY EVENT

Produced by Miller Freeman, Inc., the third largest trade show producer in North America, the OS/2 World Conference & Exhibition is presented in cooperation with the IBM Corporation and sponsored by *OS/2 Magazine*, *OS/2 Developer*, *DBMS*, *Database Programming and Design*, *LAN Magazine*, *STACKS: The Network Journal*, *Software Development* and *Dr. Dobb's Journal*.



CHOOSING THE RIGHT NEW PRODUCTS CAN BE QUITE A BALANCING ACT

The OS/2 World Products Exhibition is the only place you can get a hands-on look at the hardware and software that can make your system scream. You can pose your toughest questions as you

meet face-to-face with the major players in the industry. This world-class exhibition will feature leading suppliers showcasing applications software, communications hardware and software, databases, network solutions, utilities, multimedia tools, and more. The exhibition will run Tuesday, Wednesday and Thursday so you will have plenty of time to get the answers you need.

SEE THE FUTURE OF OS/2 TODAY

Running parallel to the technical program will be dozens of presentations by key visionaries at IBM and major hardware and software providers. Topics include: OS/2 and Workplace OS, What It All Means To You • Customizing the Workplace Shell with REXX • Exploiting OS/2 Software Motion Video • Introduction To The OSE Distributed Computing Environment • And More.

Don't miss this opportunity to learn what applications and opportunities await OS/2 users of the future, and how you can stay on the cutting edge.

Phone, FAX or mail today for more information on the OS/2 World Conference & Exhibition.



YES! Please send me info on: ☐ Attending ☐ Exhibiting

NAME _____

TITLE _____

PHONE _____

COMPANY _____

FAX _____

ADDRESS _____

CITY _____

STATE/PROV. _____

ZIP CODE _____

OS/2 World Conference & Exhibition Attn: JulieAnn Lee
600 Harrison Street, 4th Floor, San Francisco, CA 94107-9603
Phone: (415) 905-2354 Fax: (415) 905-2220

00052



*The installation and configuration of applications on workstations across LANs are becoming increasingly important tasks for network administrators. IBM's CID strategy advances recommendations for making your application easier to install and configure across a LAN. The following article describes the fundamentals of CID-enablement. It is an excerpt from OS/2 2.1 Application Programmer's Guide, reprinted with permission from Van Nostrand Reinhold. By **THEODORE SHRADER***

Enabling Your Applications for CID



Theodore Shrader

Computers interconnected over local area networks are the fastest growing segment of the PC market today. Large companies are downsizing, moving their applications and databases from mainframes to LANs. These LANs involve anywhere from a hundred to several hundred workstations. Smaller businesses are discovering the advantages that connecting the PCs in their offices has over keeping them as isolated, stand-alone workstations. With the proliferation of LANs, installing and updating applications on computers have become increasingly important and complex tasks. Feeding disks into a floppy drive for every new application is no longer a viable option. There must be a way to automate the installation and configuration of applications across a LAN efficiently and easily. Such a solution must also deal with the heterogeneous environments that exist in the marketplace today.

IBM has advanced a strategy for products called CID, which stands for configuration, installation, and distribution. Software products that are CID enabled adhere to guidelines and recommendations set forth for those applications that want to participate in the CID realm. CID enablement requires a different though not radical philosophy for the design and operation of your application's installation and configuration program. By investing the time and effort to make your product CID enabled, you will allow a net-

work administrator to install, configure, and maintain your application across all the workstations easily.

Enabling your product for the CID environment does not tie you to a specific distribution mechanism. In fact, CID-enabled applications can participate in a number of different distribution mechanisms, whether by using the LAN Configuration Utility (LCU) component of Network Transportation Services/2 (NTS/2) or through the facilities provided by the IBM NetView Distribution Manager for OS/2 (NetView DM/2) application. Novell also supports CID-enabled applications. CID-enabled products can exist in a heterogeneous environment, transparent to whichever software distribution manager your customer uses. (Software distribution manager is a generic term for any distribution mechanism that operates in the CID environment.)

Although distribution is an important part of CID, choosing a software distribution manager lies outside of the realm of influence of most software designers. Developers can, however, directly influence the way their product is designed to ensure that it can participate in the CID environment.

CID-ENABLED APPLICATIONS

Whenever a user installs an application, he or she usually performs the task in one of three ways. First, during attended installation, the user must sit at the workstation and



nurse the floppy drive with disks. The user might also be required to enter installation or configuration data into panels at different stages of the installation. This installation approach is characterized by manual, time-consuming tasks the user performs at the target workstation.

Second, during lightly attended installation, the user is not divorced from physically visiting the workstation, but the amount of work the user needs to perform is dramatically reduced. With such an approach, the user can start up the installation program for an application, possibly answer a few questions (such as what drive to install the product on), and walk away. Users can start the installation program by inserting boot disks or by executing a program from another workstation that runs on the target workstation.

Third, during unattended installation, the user does not even have to visit the target workstation. Users simply schedule the applications for installation on their target workstations through their software distribution manager. This results in the unattended execution of the installation (or configuration) programs for one or more products on the target workstation. During the course of the installation of a product, the software distribution manager reboots the workstation as required by each installation program. Successful or unsuccessful installations are reported back to the network administrator as implemented by the software distribution manager.

When the term "installation program" is mentioned, it refers to the installation and configuration programs of a product. These programs may be separate or combined, allowing one program to perform the configuration and installation of a product. Also, the terms "target workstation" and "client workstation" can be used interchangeably.

The optimal customer solution would be to have all applications installable in an unattended fashion. For example, the applications on all the workstations in an office could be installed and configured overnight from a

remote site and be ready for use in the morning. However, unattended installation is not always possible, since some workstations may not have the support software to retrieve or communicate information across a LAN, for example. Nevertheless, getting products to a point where users can install and configure them in a lightly attended or unattended fashion will save a great deal of work, lower LAN maintenance costs, and ensure that products and upgrades are distributed to the target workstations on a LAN quickly and efficiently. A product with all these characteristics will be prized by network administrators.

CID-enabled products can operate in lightly attended and unattended modes if they support two installation techniques. With the redirected installation technique, the installation program for the application must be able to retrieve its code images (in such forms as compressed files or disk

```
\CID
  \CLIENT (stores the client command files, such as those
            used by the LAN CID utility)
  \DLL (stores .DLL files, generally those used in the EXE
        directory)
  \EXE (stores executable programs files)
  \IMG (stores the disk images or product files—each sub-
        directory pertains to a specific product)
    \PROD1
    \PROD2 (and so on)
  \CSD (stores the Corrective Service Disk or Service Pak
        images for a product—each subdirectory pertains to a
        specific product)
    \PROD1
    \PROD2 (and so on)
  \LOG (stores the log and history files for a product—
        each client subdirectory pertains to a specific product)
    \PROD1
    \PROD2 (and so on)
  \RSP (stores the response files for a product—each
        client subdirectory pertains to a specific product)
    \PROD1
```

Table 1. Directory structure for products available on a code server



images) from any drive, whether a network redirected or CD-ROM drive. A network redirected drive physically exists on another workstation on the LAN, but to a client workstation, it looks like a local drive because of the software installed on the client. Users should not have to use the floppy dri-

ves on their workstations to install an application. To a large extent, redirected installation means that a product can be installed from across a LAN.

With the response files technique, the response files, which are ASCII files with specific keyword and value syntax, replace the need for the user to input

installation and configuration values manually at install time. Response files remove the need for all users to know what the installation and configuration parameters mean. The LAN administrator can provide a specific response file that tailors parameters, such as the maximum database buffer size, for a group of workstations or a single workstation. A CID-enabled product will use a response file for its installation and configuration processes. Response files do not eliminate the need for a product's regular installation program. Users should still be able to install and configure their product through a windowed interface.

Once a product supports redirected installation, response files, and a few additional command-line parameters for its installation program, the application has suited up as a player in the CID arena.

There's a difference between a product's installation program and its regular program, which may consist of several programs—the total of which is called the application. The installation program is used to install or configure the program from images or files on a disk or code server. It is typically run only once or very infrequently to establish or configure a program on a workstation. The user invokes the application program files to use the actual product.

The configuration of a product need not take place at installation time. For example, users may want to change their configuration values for tuning purposes after installing and testing applications across their LAN network.

CODE SERVER AND ITS DIRECTORIES

The term "code server" is used frequently when describing how an installation program can retrieve the response and product code files, whether in disk image or compressed format, for installation on a target workstation. A disk image is a binary file containing the contents of an entire disk. Compressed files are created by compaction programs. The code server acts as a central repository that the installation program can

Rhetorex Voice Processing boards make CTI a reality.



If you're asking "what's CTI," you're missing one of the hottest new technologies going.

Computer Telephony Integration links PC-based computer applications to the telephone network, providing voice/fax mail, interactive voice response, and other telephony applications.

Interested? Maybe you're already developing a CTI application. Then it's time to discover Rhetorex.™

The fact is, when it comes to multi-port voice processing components, no one offers more value than Rhetorex.

With Rhetorex voice platforms, you'll get state-of-the-art DSP-based technology, designed from

the ground up for reliability. Sophisticated DSP algorithms give you unparalleled performance. Best of all, enhancements to algorithms are made via software upgrade.

And Rhetorex products help you develop applications quickly. All components include device drivers, a straightforward API, and sample programs. Friendly, free technical support is available at any stage of development.

For the best value in CTI technology—from our 2 and 4 port voice processing boards and fax boards, to our 24-port platform—then give Rhetorex a call. (408) 370-0881. And start making CTI a reality.



RHETOREX

Rhetorex, Inc.

200 E. Hacienda Avenue
Campbell, CA 95008-6617
Tel. (408) 370-0881
Fax (408) 370-1171

All trademarks identified by the ™ symbol are trademarks of Rhetorex, Inc.
All other trademarks belong to their respective owners. © 1993 Rhetorex, Inc.

Circle Reader Service Number 39

access using redirected installation.

One such recommended directory structure for products available on a code server is shown in Table 1.

The response files would have names like NODE0001.RSP and NODE0002.RSP, for example, where the file names are the same as the client node names. The file-name \CID\RSP\LS30\NODE0001.RSP would indicate to the user that the response file was for the target workstation named NODE0001, and the response file was specific to the LAN Server 3.0 product.

The command files in the \CID\CLIENT\ directory are used as part of the software distribution manager target (or client) installation process to identify the products that need to be installed, the order of installation, and the commands needed to complete the installation. The software distribution manager the user chooses may require its own unique directory structure format. With support for response files and redirected installation, your application need not concern itself with the functions of the software distribution manager, yet it will still be able to participate in the CID environment.

REDIRECTED INSTALLATION

To design your application to support redirected installation, your installation program will need to support two functions.

First, it will need to support the ability to transfer the images or contents of the program installation disks to a hard drive or to another medium. This allows network administrators to place your product images on their code server.

Second, it will need to support the ability to retrieve the disk images or program files from a network, redirected, or nonfloppy drive during the installation process.

Redirected installation does not mean your program cannot be installed from a floppy drive. In fact, being able to install from the A or B drive is still of paramount importance to the large base of stand-alone workstations. Supporting the installation of your pro-

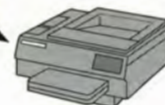
gram from a floppy drive is still a primary requirement for most applications. Yet by extending your installation program to support redirected installation also, you have extended the reach of your program to the stand-alone and LAN environments and made your application more attractive to LAN

administrators and users.

An application could use an operating system utility program, such as XCOPY, to transfer program files from its product disks to a code server. It also could use XCOPY to transfer files from a network or redirected drive to the target workstation. You don't need to build a



SINGLE KEY-STROKE SCREEN PRINTING



CLIPBOARD
VIEWER



SCREEN
SAVER

SCREEN CAPTURE
.WPG .MET .BMP .PCX .TIF .GIF .PCT



- Quality **Color or Black & White** copies of color Desktop Images.
- Copy any portion of any image on the Desktop.
- Screen Saver prevents monitor burn-in.
- Date & Time Display for Time-Stamping images.
- LAN installable. Complete support for LAN attached printers.
- Economical! 4 Utilities for 1 Price! Entity licensing available.
- The most stable and reliable product available.
- Includes both OS/2 2.1/2.0 (32-bit) and OS/2 1.3 (16-bit) versions.

THE LEADER: PrntScrn™ Screen Image Utility for OS/2



TM

MITNOR Software
Phone: (918) 357-1628
Fax: (918) 357-2869

SCREEN IMAGES IN OS/2 DEVELOPER ARTICLES ARE CAPTURED USING PRNTSCRN, COURTESY OF MITNOR SOFTWARE

Circle Reader Service Number 40



special function to accomplish this.

Applications, including those not specifically designed to be CID enabled, have already started to support redirected installation. Typically, this takes the form of allowing the user to copy the contents of the program disks to a hard drive or network partition. Afterward,

when the user starts the installation program, it will prompt the user at the beginning to ask for the drive and directory path from which to retrieve the program files. These programs also ask for the target partition and directory in which to install the application.

More sophisticated installation

programs will copy the disk images to predetermined directories on a target disk partition. These programs will even create the directory structures if they do not exist. Whether the user performs an XCOPY on the disks or uses the installation program to transfer files, supporting redirected installation helps move users away from having to perform attended installation.

RESPONSE FILES

The main purpose of response files is to provide predetermined answers to all the possible installation and configuration prompts that the installation program may ask the user. This frees the user on the target workstation from having to know the parameters, their ranges, and so on. Someone, perhaps the network administrator or an expert with the application who knows how to tune the parameter values for optimal performance, can build the response file for the client workstation before the installation program is even invoked.

Again, response files do not replace the need for panels that prompt the user for values or ask the user to select from a list of choices. First time users may prefer this route with its context-sensitive help instead of leafing through a manual or installation guide. Response files may be useful, but are not a requirement, for stand-alone users who are not likely to install the program more than a few times.

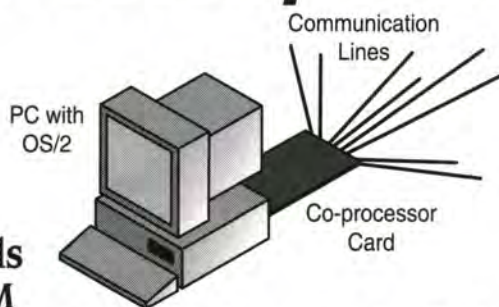
There are some programs, very few in number, that do not require the user to specify any installation or configuration parameters. (Even most of today's entertainment software will ask you to choose the sound cards and video displays on your system to confirm or override any default selections it may have detected.) The net result is that without support for response files, your program may not be able to support lightly attended installation fully. For programs that require parameters, your application certainly will not be able to support unattended installation.

Response files are ASCII files that

Cut your development time

Quadron's OS/2 development tools team up with IBM

ARTIC co-processor cards to give you flexibility, communication power, and development speed.



Use Quadron software and ARTIC cards to gain extra ports and enhance host PC performance by moving protocol processing to the cards. Develop standard and custom communications for async, BISYNC, HDLC, X.25, LAP-B, or your own special protocol. And if you need more than one protocol, no problem—just run them at the same time on the same card.

Programmers feel right at home with our software. Just program in standard C, and use our messaging API to link OS/2 threads with

tasks on the card. Then check it out with our symbolic debugger and real-time analysis tools in OS/2 windows.

Our easy-to-use software serves applications as diverse as telephone switching, financial transactions, satellite communications, and process control. If any of this sounds like a match for your needs, call us today.



Quadron

209 East Victoria Street
Santa Barbara, CA 93101

Fax: 805-966-7630

805-966-6424



Trademarks are the property of their respective owners.

Circle Reader Service Number 38

a user can freely edit. Each line in the file has a maximum length of 255 characters. The most basic unit of a response file is the **keyword=value** pair. For example, the following entries in a response file specify the configuration values for the maximum number of database connections and the database buffer size:

```
DB_MAX_CONNECTIONS = 254
DB_BUFFER_SIZE = 64
```

Response files can contain keyword assignments that are specific to both installation and configuration. It is up to the installation program to determine if such a distinction needs to be made. This means that the program could, for example, accept and process all the installation and configuration parameters in a response file when the program is first installed. If the user or administrator runs the installation program again and if the program detects that the program already exists on the target machine, the installation program would still use the response file. It would, however, ignore those parameters that only pertained to installation. If the user specified a reinstallation of the program, the installation keywords would then be used.

CONCLUSION

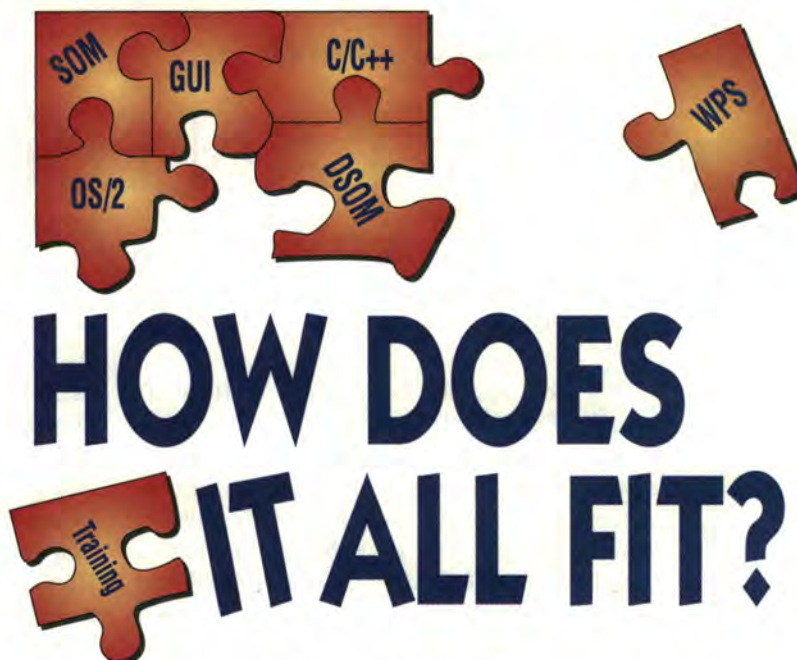
With the explosive growth in LANs, users are demanding an easier way to configure, install, and distribute their applications across all the workstations on their LANs. The CID strategy provides an easy way in which to enable applications to participate in the LAN environment. You may find it odd that many of the features of making your product CID enabled are presented as recommendations rather than requirements. CID was designed to allow application developers as much leeway as possible in designing their products while still being able to participate in the CID environment. By including these suggested functions in your design, you can make your pro-

gram more attractive to network administrators and users, as well as make it easier to learn and use.

ACKNOWLEDGEMENTS

The author would like to thank Tom Edel and fellow book co-authors, Jody Kelly, Craig Swearingen, and Dawn Bezviner, for their time and assistance.

Theodore Shrader, IBM Personal Systems Line of Business, Austin, Tex., joined IBM in 1989 to work on graphical interfaces to the Database Manager product. He has worked in graphical development since then, which included work on local area network programs. He holds a B.S. in computer science from the University of Texas at El Paso.



Developing today's complex applications can be quite puzzling. It requires more and more knowledge and training. A complete understanding of the capabilities of OS/2 and the establishment of programming standards will help you produce applications that give your business a competitive edge in the market place. Ask the OS/2 specialists...

We can help you create the most productive environment for your application developers through training, consulting, and programming services. Our object-oriented software engineering and GUI design techniques are the keys to make your application development a success.

Benefit from our proven experience!



621 NW 53rd Street, Suite 240
Boca Raton, FL 33487 USA
1-800-4SE-INTL or (407) 241-3428
Fax: (407) 278-3919

SES GmbH Berlin
Koenigsallee 49
14193 Berlin, Germany
+49 30 826 40 63
Fax: +49 30 825 30 14

Circle Reader Service Number 41



LAN

With the increase in demand for workstation disk storage by evolving operating systems, efficient system planning and hardware investment protection are more critical than ever. The distributed feature option of OS/2 Communications Manager/2 (CM/2) v.1.1 can exploit the functions of a LAN-based file server, resulting in significantly reduced storage requirements and efficient system maintenance. Here, we overview the distributed feature, show examples of installations, and discuss reducing the effects of LAN connectivity losses.

By STEPHEN SMITH, BRUCE KOHLER, and ADAM KING

Increasing LAN Efficiency with CM/2 Distributed Feature

The distributed feature option of CM/2 allows multiple distributed LAN workstations to access and run Communications Manager from files stored on a single file server workstation. The distributed workstations use a LAN requester facility to connect to a designated drive on the file server as if the drive (and the files) were stored locally.

Because distributed workstations require that only a few CM/2 files are installed on a user's workstation, the amount of fixed disk space necessary to run CM/2 is significantly reduced. Workstations that typically would require 10MB to 15MB for Communications Manager files (depending on their specific configurations) now need only 2MB or less with the distributed feature arrangement.

Because the number of distributed workstations connected to a single file server is limited only by the physical LAN installation and the server's capacity, the distributed feature option can result in a big reduction of LAN workstation disk requirements, extending the useful life of existing limited-storage workstations.

In addition to reduced disk storage, the Distributed Feature option can greatly simplify LAN installation and

maintenance and improve network availability. The CM/2 product files only need to be installed on the server; this greatly reduces the cost of distributing changes and tracking maintenance levels across multiple workstations. If maintenance is needed, it is done at the server, and the workstations get the benefit of the changes with a minimal impact, instead of a full CM/2 reinstallation. The distributed feature also allows you to install multiple service levels and national language support (NLS) versions of CM/2 on a single file server, where they can be accessed by workstations as needed.

INSTALLATION AND CONFIGURATION

The CM/2 Distributed Workstation Feature is licensed as a distributed feature of the OS/2 Communications Manager/2 base license. CM/2 provides the code required to install a file server and create unlimited distributed feature workstations from the server. Each installed workstation needs a separate distributed feature license.

File Server Installation. There are two ways to set up a file server to support CM/2 distributed feature workstations, depending on whether or not CM/2 actually executes on the server.

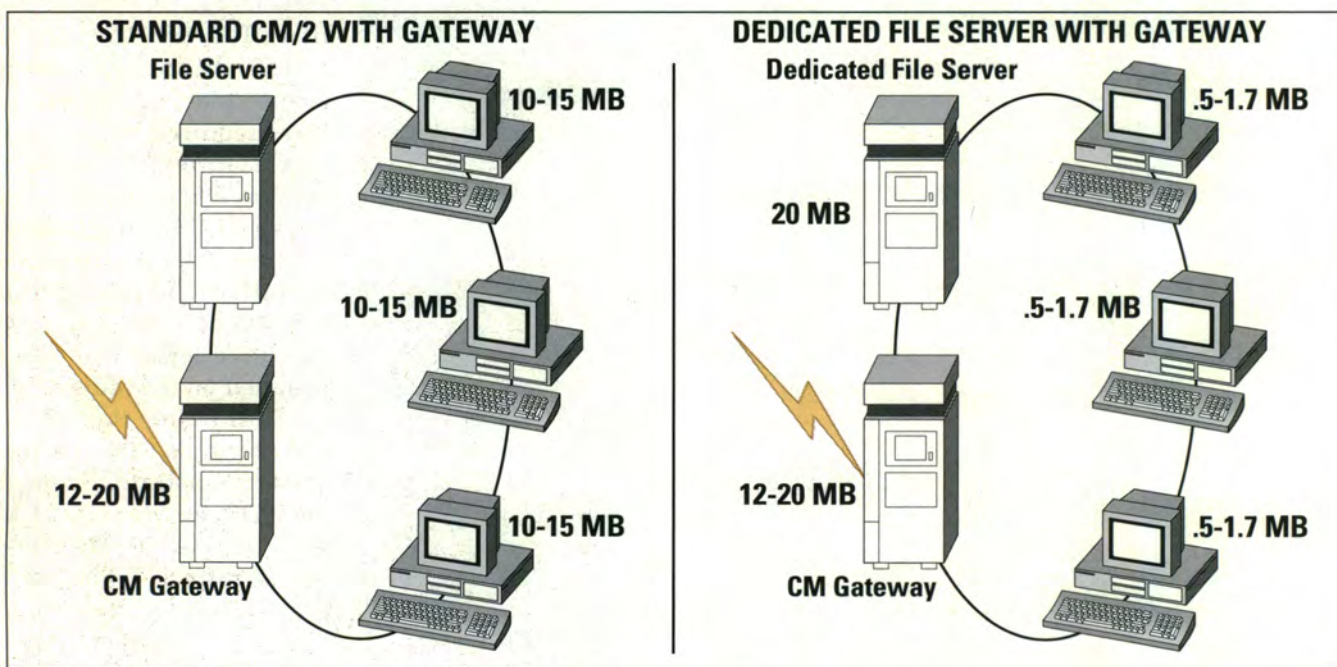


Figure 1. Comparison of standard CM/2 installation vs. dedicated file server with distributed workstations

If the LAN administrator does not intend to run CM/2 on the file server machine, the file server is called a dedicated file server, and storage requirements are fixed at 20MB. Additionally, because a dedicated file server does not actually run CM/2, you do not need to install OS/2 on the file server. You can use any LAN-based file server (IBM LAN Services, Novell NetWare, Banyan Vines, and so on). You can also use the CM/2 CMIMAGE utility to install a dedicated file server.

If you want CM/2 to execute on the server machine, share the CM/2 product files between the executing CM/2 and the attached distributed workstations to limit disk requirements. This is a communications file server setup, and the installation is slightly different. The administrator installs CM/2 using the CMSETUP utility, specifying the server's operational configuration. Next, the CMSERVER utility is used to prepare the operational CM/2

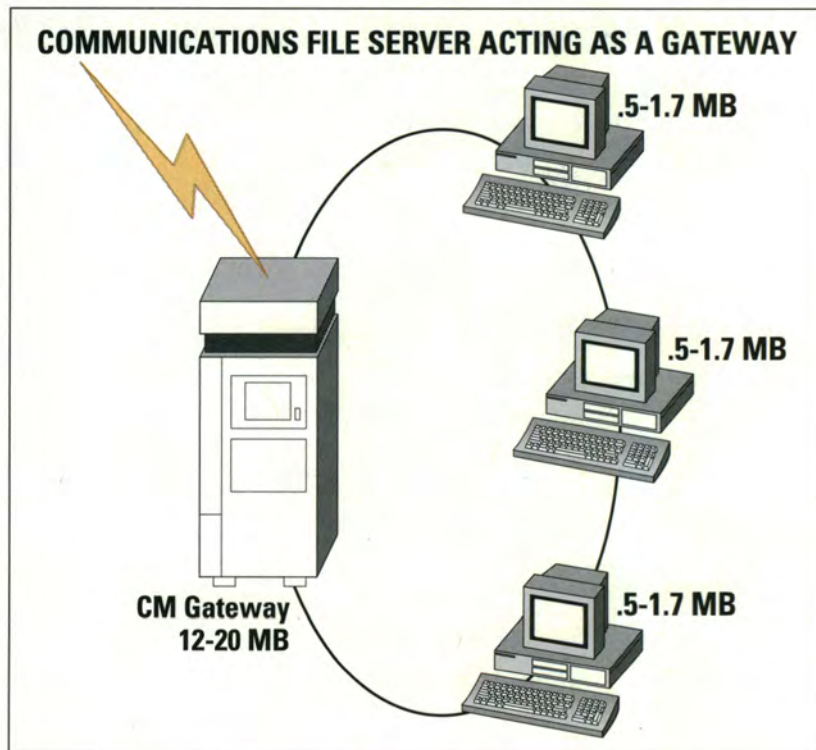


Figure 2. Distributed installation with SNA gateway



Function	Base CM/2 Workstation	Distributed Workstation
Interactive Configuration	YES	NO
Runtime reconfiguration		
-Emulator	YES	YES
-SNA/APPC	YES	YES
SNA Services/APIs		
APPN	YES	YES
CPI-C	YES	YES
LUA	YES	YES
SRPI	YES	YES
LU 6.2	YES	YES
Common Services	YES	YES
Emulators		
3270/5250	YES	YES
ASCII*	YES	YES
Connection Types		
LAN	YES	YES
COAX	YES	NO
TWINAX	YES	NO
Asynchronous	YES	NO
Switched lines	YES	NO
Leased lines	YES	NO
Connectivities		
802.2	YES	YES
SDLC	YES	NO
ASYNCR	YES	NO
NETBIOS	YES	NO
X.25	YES	NO
ISDN	YES	NO
Gateway	YES	NO
APPN	NN/EN	EN only
ACDI, ACDI redirection	YES	YES

*The Softerm Custom Plus emulator by Softronics Inc. is shipped with Communications Manager/2

to also act as the file server.

This mode only requires as much disk space as is needed by the operational configuration. However, the distributed workstations can only use files that are installed by the operational CM/2. Therefore, you need to ensure that a superset of all features required by the distributed workstations is installed on the server. Server disks typically require 12MB to 20MB, depending on the features that are installed. This configuration requires that OS/2 and any LAN-based file server that uses OS/2 is installed on the file server.

Following the setup of the CM/2 product files on the server, the administrator then provides an ALIAS for the CM/2 libraries and provides the distributed workstation users with READ and EXECUTE authority (assume that a LAN requester product is installed already on the distributed workstations.)

DISTRIBUTED WORKSTATION INSTALLATION

The network administrator creates Communications Manager configuration files that are necessary for the installation of distributed feature workstations and stores them centrally on the server.

Using those configurations, workstations are able to access the streamlined CM/2 installation to get the few files necessary for running CM/2. As an alternative, the network administrator can install the workstations remotely, using the configuration, installation, and distribution (CID) capability of OS/2, via the NetView Distribution Manager/2 product.

Although a distributed workstation does not support the CM/2 PM-based interactive configuration utility, the user still

Table 1. Standard and distributed workstations function availability

can perform run-time configuration changes, such as the modification of file transfer profiles and keyboard, pop-up keypad, and color customization.

In addition, users can modify any Systems Network Architecture (SNA) configuration parameters that exist in the CM/2 network definitions file (NDF) by using an ASCII editor and then running the CMVERIFY utility. If centralized configuration management procedures are in place, the user must send these modified files back to the central repository.

Run-time Functions. Distributed workstations have essentially the same communications services and APIs as a standard CM/2 LAN-attached workstation. These services include IEEE 802.2 for communications on the LAN and ASYNC support for asynchronous communications device interface (ACDI) and ACDI redirection. You can obtain off-LAN access via a CM/2 gateway machine, a CM/2 advanced peer-to-peer networking (APPN) network node, or through another network communications controller (37x5, 3174, and so on). A distributed workstation cannot function as a gateway or as an APPN network node.

Table 1 shows a comparison of the available functions between a standard Communications Manager workstation and a distributed workstation.

Other than exception handling for a LAN connectivity loss, run-time execution at a CM/2 distributed feature workstation is transparent to the user. If CM/2 detects the loss of connectivity to the server as a result of a break in a physical or logical connection, it notifies the user that it cannot continue and terminates. It then performs a cleanup to prepare itself to be restarted by the user when connectivity is later restored.

LAN Connectivity Losses

Communications Manager/2 is a complex OS/2 application with dozens of processes, subprocesses, and threads that run concurrently. Further, in a distributed feature environment, each workstation typically will have 60 or more files opened from the server at any given time. Loss of connectivity to the server in this environment can be catastrophic. The CM/2 distributed feature includes a LAN loss detection routine to minimize the effects of connectivity losses and facilitate recovery from such outages.

LAN connectivity losses can occur for various reasons. Most typical outages are physical in nature: severed or removed LAN cables, power outages, or disconnected remote drives on the distributed workstation. Other reasons for connectivity losses include intentional logoff from the server by the user, stopping the requester program while processes that access open server files are running, and normal or abnormal termination of server programs. All of these conditions can result in failures, ranging from unrecoverable page faults to traps, as processes (either active or dormant) that were loaded prior to the outage attempt to run in response to an internal or user-driver request. Because the operating system doesn't keep track of all the processes and subprocesses loaded by each application, the user may not be fully aware of the dormant processes still loaded in the system and may believe that the application has been terminated. These dormant processes will be subject to the same failures once the user attempts to execute the application again.

The effects of connectivity loss result from the OS/2 classification of network drives as nonremovable, the OS/2 optimization of shared dynamic link libraries (DLLs), and the recovery techniques employed by network file systems.

The nonremovable classification assigned to network drives allows the OS/2 loader to assume that executable code loaded from a server will always be accessible. This results in reduced swap file growth by allowing the executable code to be reloaded from its source rather than copied to the swap file. This is why .DLL and .EXE files are locked while they are executing.

The OS/2 optimization of sharing DLLs further complicates LAN loss recovery. If the application being loaded accesses a DLL that is already being used, the OS/2 loader assumes that the DLL can still be read from the disk. This can pose a problem when the application is local, and the DLL is located on the file server. The locally installed application will load successfully and then experience the effects of network loss when it attempts to call the shared DLL, which is no longer accessible.

To complicate this scenario, the application may successfully call the DLL when a portion of that executable code is stored in the machine's memory. On a lightly loaded machine, the application could execute for hours after a connectivity loss to the server and

```
if normal termination
    delete current PID from shared list
else
    if network access lost
        request display of failure message
        disable display of fatal trap messages
        for all PIDs in the shared list
            DosKillProcess()
    exit
```

Figure 4. LAN loss algorithm

may not experience a full loss until it is catastrophic.

Most network file systems use similar methods of network drive recovery. Any lost network drive, once reconnected, has the internal representation of an entirely different drive with the same files. The file system loader, when called on to reread an executable file, detects that the file cannot be read from its original location and fails. Although the network drive appears the same to the user (same drive letter and files), the loader knows that the original cannot be accessed. Consequently, any process that is loaded from the reconnected network drive will fail as if that drive had never been reconnected.

The LAN loss function provided by CM/2 attempts to reduce the effects of connectivity loss. It does this by using the OS/2 capability of registering a subroutine to be run at program termination. This exit list processing is supported for OS/2 1.3 and OS/2 2.x. The exit list routine is stored in a DLL that is always installed locally to ensure access in the event of a LAN connectivity loss. Each Communications Manager/2 process and all user processes that access Communications Manager/2 via an API link to the DLL directly or indirectly.

The DLL initialization routine registers an exit list routine in all the processes and stores the process identifiers (PIDs) in a shared memory area. As each process exits, the routine is notified if the process has terminated normally or abnormally.

In the event of abnormal termination of a CM/2 process, CM/2 checks the accessibility of a file known to reside on the file server; if the file cannot be accessed, the connection is assumed to be lost. Abnormal termination in the event of LAN connectivity loss is usually the result of a fatal page fault. This basic algorithm is described in Figure 4.

Communications Manager/2 LAN loss processing proceeds as follows:

1. A request to display a message indicating that CM is terminating due to LAN loss.
2. The shared memory list of process identifiers is accessed and a **DosKillProcess()** is issued for each one.
3. The OS/2 fatal page fault message is suppressed. If the loss occurs in a system-wide debugger environment such as the OS/2 debug kernel, the fatal page fault is seen as trap 000E.

All user processes are left running unless they attempt to access unavailable .EXE or .DLL files. In this event, they are stopped individually by OS/2 as they fault. If the network is restored using the methods employed by most network file systems, the user must stop each process before attempting to restart CM/2. CM/2 provides the productivity aid **CMPROC.EXE** to help identify and terminate user processes that have attached to CM/2.

Very few network file systems provide an active notification of LAN loss. As an alternative, you can poll for file server accessibility or adopt a passive detection system, as CM/2 does. Polling wastes CPU cycles and network bandwidth checking for a low frequency event and fails to rule out the opportunity for network loss failure without detection completely.

SAMPLE DISTRIBUTED INSTALLATIONS

Three examples of distributed feature installations highlight the different installation approaches that are available.

Dedicated File Server with Distributed Feature. In Figure 1, the example compares a standard CM/2 installation with a distributed feature installation using the same equipment. We compare disk requirements for each type of installation.

In Figure 1, a LAN file server is assumed to already exist on the LAN.

Communications File Server Acting as a Gateway. The example in Figure 2 shows a distributed feature network in which the file server also acts as an SNA gateway attached to a wide area network.

Multiple Language and Version Installation. The example in Figure 3 shows how you can install multiple national language versions or CSD upgrades of CM/2 on a single file server and access them across the LAN. A multiple language setup is useful for sites that require multilingual

REFERENCES

Communications Manager/2 Version 1.1 Information and Planning Guide, (SC31-7007-01, 79G0459)

Communications Manager/2 Version 1.1 New Features, (GG24-4142-00)

installations, such as corporate training facilities or travel reservation companies. If you have multiple CSD levels available concurrently, it facilitates service-level testing prior to distribution throughout the network.

Stephen Smith is a senior associate information developer with IBM's Networking Software Division, Research Triangle Park, N.C. He has been with IBM since 1983 and has worked with the Communications Manager product since 1990. Smith attended Barry University.

Bruce Kohler is a senior programmer in the Systems Software and Development Services organization of the Networking Software Division, Research Triangle Park, N.C. Most recently, he was responsible for

The passive method employed by CM/2 is more efficient but leaves the possibility of experiencing the effects of LAN connectivity loss without detecting the source of the problem. Failure without detection can occur when network access is lost and restored while CM/2 is only executing code that is stored in memory. The result is a series of inexplicable traps. To restore normal operation, the user can run **CMPROC.EXE** to stop all CM/2 and user processes or reboot. Most occurrences will be detected, CM/2 will be stopped, and the user can stop all transaction programs, possibly saving user data. When file server access is restored and all processes are stopped, CM/2 can be restarted without a complete system reboot.

the testing of the distributed feature option of OS/2 Communications Manager/2 and has held other lead testing roles for CM/2 since 1990. Kohler joined IBM in 1974 and holds a B.S. in computer science from Pennsylvania State University.

Adam King is a developer with IBM's Networking Software division in Research Triangle Park, N.C. He has been with IBM since 1991. King holds a B.S. in computer science from Furman University and an M.S. in computer science from Clemson University.

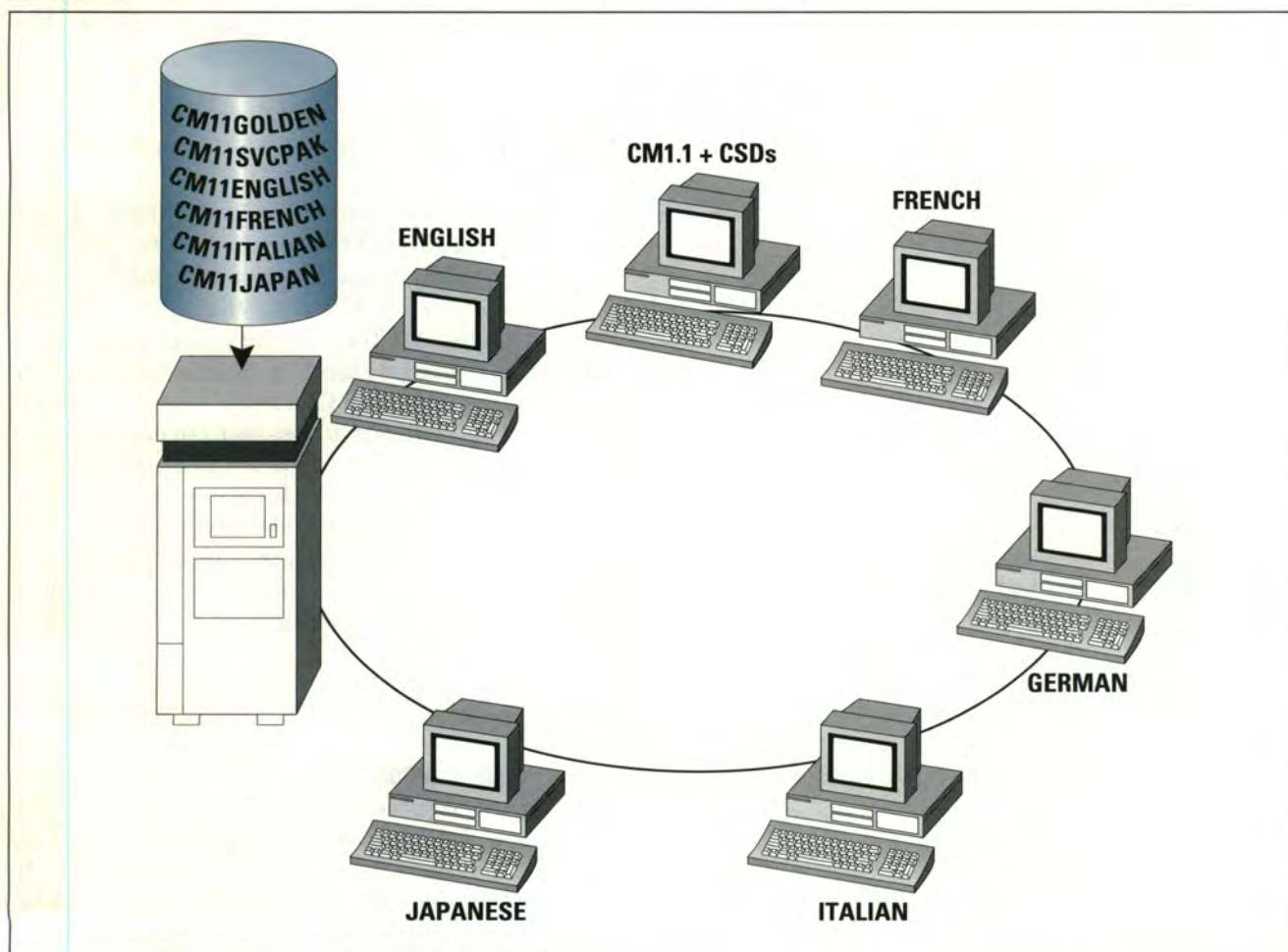


Figure 3. Distributed national language installation



New Tools and Utilities



TOOLS

Footprints 2.0. A real-time trace facility for 16-bit and 32-bit applications running OS/2 2.x, Footprints 2.0 allows developers to trace and debug code in real time, without the need to stop or slow the program down, according to the company. Footprints only traces when embedded trace switches are activated from an external PM interface. It can also create time-stamped archive files of traced data and paths of execution. Once the developer-defined trace statements are compiled into the code, their switches can be turned on and off without recompiling. Footprints 2.0 is priced at \$99.00.

Hardy Software Systems Inc.
Phone: (713) 871-1448, Fax: (713) 871-1449

GpfRexx. GpfRexx is a visual programming environment that allows OS/2 users to design, test, and generate Presentation Manager graphical user interfaces for programs in REXX. A point-and-click interface design tool set, GpfRexx lets even users without detailed system-level knowledge design, test, and generate user interfaces that take advantage of OS/2 features, the company says. Users point and click to design interface windows, menus, and OS/2-style, context-sensitive and extended help. GpfRexx requires OS/2 2.1, a mouse; at least 6MB of available hard disk and 8MB of RAM are recommended. GpfRexx costs \$247.50.

Gpf Systems Inc.
Phone: (800) 831-0017 or (203) 873-3300,
Fax: (203) 873-3302

VisPro/REXX 2.0. VisPro/REXX Bronze 2.0 and VisPro/REXX Gold 2.0, which make up version

2.0 of the VisPro/REXX product set, are visual programming tools for building OS/2 2.x GUI applications. The new version includes support for creating visual REXX macros, adding objects using IBM's SOM, and accessing the Q+E Database Library. VisPro/REXX Gold 2.0, which replaces the original product, features built-in entity/relationship database diagramming. VisPro/REXX Bronze 2.0 is priced at \$99.00; VisPro/REXX Gold 2.0 is \$299.00.

HockWare Inc.
Phone: (919) 380-0616, Fax: (919) 380-0757

Enhanced Printer Driver Tool Kit for OS/2. The Enhanced Fax Printer Driver helps users write applications that create and send fax documents, or add fax to existing applications. By printing ASCII text to the printer driver, users can create and send fax documents, according to the company. Command lines convert text using different fonts, add cover pages and header lines, and automatically send the fax document. Users can broadcast faxes to lists of recipients. The printer driver is compatible with SofNet's FaxWorks OS/2 products and Keller Group's PMfax products. Enhanced Printer Driver Tool Kit is available for \$295.00.

Keller Group Inc.
Phone: (612) 429-7273, Fax: (612) 653-1987

WidgetKit/CUA'91. WidgetKit/CUA'91 adds the CUA'91 controls to the standard controls available for use in WindowBuilder Pro/V. WindowBuilder Pro/V enables programmers to build user interfaces interactively and rapidly, according to the company. WindowBuilder Pro/V with WidgetKit/CUA'91 provides a CUA interface development environment for



Smalltalk/V OS/2. WidgetKit/CUA'91 has an implementation of the IBM CUA'91 controls, including notebook, container, slider, value set, and spin button. It is priced at \$295.00.

Objectshare Systems Inc.
Phone: (408) 727-3742, Fax: (408) 727-6324

CA-SuperProject for OS/2. A multiple project management tool, CA-SuperProject exploits such OS/2 features as the Workplace shell, preemptive multitasking, and enhanced memory management. According to the company, CA-SuperProject contains ease-of-use features, including Help Assist mode, experience modes, pop-up forms, and Project Manager's Assistant; views, including four outlines with Gantt and histogram charts and crosstabs; project modeling, featuring effort-driven scheduling, flexible data and precedence constraints, and work breakdown structure; and resource management and costing. CA-SuperProject for OS/2 is compatible with Windows 3.1 and OS/2 2.1 and is bundled with the Windows version at a price of \$649.00.

Computer Associates International Inc.
Phone: (516) 342-5224, Fax: (516) 342-5329

Automated Test Facility 3.0. Automated Test Facility (ATF) 3.0 tests client/server applications written under Windows and OS/2. ATF's architecture, which features a central point of control, is capable of testing multiple workstations interacting simultaneously. This enables ATF to test client/server scenarios for concurrent requests and lockout, as well as more complex cases in which the results of one user's activities may trigger the tasks of other users. With this new version, users can incorporate legacy system testing through ATF's new support for 3270 terminal emulation. Other features include a capture/playback facility and scripting language. ATF prices, depending on testing configuration, start at \$18,000.00; this provides the central point of control and the ability to test applications on one workstation. Each additional workstation costs \$2,000.00.

Softbridge Inc.
Phone: (617) 576-2257, Fax: (617) 864-7747



UTILITIES

CPU Monitor Plus. CPU Monitor Plus is a utility that provides graphical and text displays of CPU Utilization for processes and threads and RAM utilization estimates for processes including detailed analysis of allocated, in-use, private, and shared totals. It also allows users to change execution priority or suspend individual threads in OS/2 Presentation Manager applications. CPU Monitor Plus is priced at \$129.95.

BonAmi Software Corp.
Phone: (508) 371-1997

Sterling DMS/IB for OS/2. Sterling DMS/IB is a backup utility that supports extended attributes including long file names; works with FAT or HPFS or both; runs as a background process; runs from command line or Presentation Manager text box; and supports SCSI-2 tape devices. Sterling DMS/IB is priced at \$180.00.

Sterling Software
Phone: (916) 635-5535, Fax: (916) 635-5604

OS/2 CALENDAR

March 15-17	Software Development, San Jose, Calif. (800) 441-8826
April 19-21	OS/2 Technical Interchange, San Francisco (407) 982-9353
May 2-5	PEN EXPO, Boston (508) 649-4200
May 8-12	IDUG International DB2 Users Group, San Diego (312) 644-6610
May 23-26	Comdex Spring, Atlanta (617) 449-6600
May 23-27	DB Expo, San Francisco (415) 966-8440
June 23-25	Personal Communications & Computing, Chicago (202) 457-8700
July 19-22	OS/2 World, Santa Clara, Calif. (415) 905-2354

ODBC or NOT?

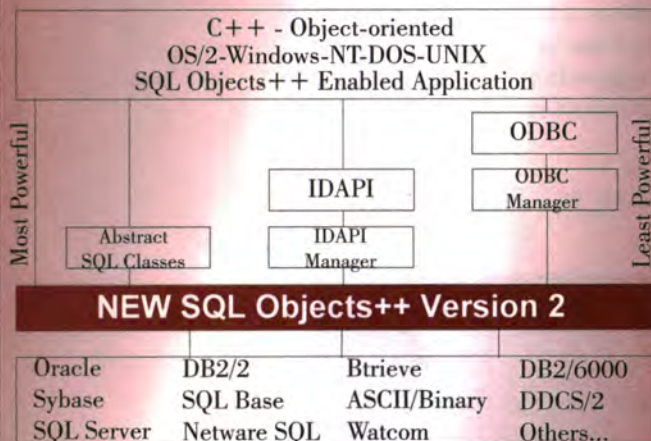
That's the question.

Ever wonder if ODBC really means "Only Does Basic Connectivity" or if Q+E stands for "Questionable + Expensive"? If so, join the crowd of programmers who have discovered the *true* meaning of Open DataBase Connectivity.

SQL Objects++ is the answer.

The exploding Client/Server market has everyone searching for a database access method to hang their hat on. Unfortunately, all of these methods limit you to the lowest common denominator. With the release of version 2 of SQL Objects++ "you can have your cake and eat it too."

Don't base your application or future on a "one access" solution. Use a solution that allows you to access your database any way you like. Use a solution that is designed for performance. Use a solution that handles multiple operating systems with the same set of code. Use SQL Objects++.



We have a lot more to tell you but there's not enough room. So, pick up the phone right now and call **800-876-6585** for more information. Better yet, call and ask about our "No Risk (Really!) Guarantee" and our "\$99 Test Drive" option.

Objects++TM

SOFTWARE CORPORATION

47 Stonewall Street Cartersville, GA 30120

(404) 382-6585 - fax (404) 382-6374

SQL Objects++ and Inherit The Power are trademarks of Objects++ Software Corporation. All company and product names are trademarks or registered trademarks of their respected owners.

ADVERTISER INDEX

Reader Service	Company Name	Page
45	Adaptive Research & Design	93
52	Borland International.	COV4
46	Burton Systems Software	93
13	Compuware	25
28	Creative Systems Programming	60
20	Development Technologies, Inc.	37
51	Digitalk.	COV3
47	dSoft Development	94
37	Essex Systems, Inc.	63
	Game Developer.	54
23	Globalink, Inc.	43
15	Gpf Systems Inc.	28
26	Hardy Software Systems	59
30	Hilbert Computing	62
5	HockWare	11
	IBM Canada DB2/2.	51
	IBM Canada	21
	IBM GIK.	9
	IBM Technical Interchange.	33
	IDUG Conference	57
48	Impact Software	94
29	Indelible Blue, Inc.	62
19	Intersolv	35
32	IRMware Services	62
17	Kaseworks	31
36	Keller Group.	63
49	Levine Computer Consultants	95
21	MHR Software & Consulting.	39
40	Mitnor Software	69
2	Micro Focus	2
43	Object ++	80
27	The Object People	60
34	Op-Tech Data Processing.	63
	OS2 Applications Directory	53
	OS2 Magazine	27
	OS/2 World Conference & Exhibition.	64-65
	Pen & Portable Computing Conference.	41
61	Perez Computing Service.	63
3	Programmer's Paradise.	4
38	Quadron	70
35	Real Time Technologies.	63
39	Rhetorex Inc.	68
6	Rimstar Technology.	13
41	SE International	71
44	ScheduPerformance.	92
50	Softbridge, Inc.	95
24	SofTouch Systems	45
	Software Development	49
7	Synaptec	15
9	Token Technology.	17
60	Token Technology.	62
12	Van Nostrand Reinhold.	23
31	Walnut Creek CDROM.	62
1	Watcom C for IBM PC's	COV2-1



Visual programming is currently a hot issue in the desktop marketplace. This article demonstrates a visual programming tool in a sample-intensive manner, based upon a recent job assignment by one of the authors. By MARK A. BENGE, JOHN BOEZEMAN, and JAMES PIRKLE JR.

Visually Sampling DB2/2

As OS/2 gains acceptance in the marketplace, tools that promote application development for the operating system continue to arrive on the scene. One such tool uses a visual interface for program construction. Visual programming tools are designed to encourage rapid creation of applications. Our goal is to provide a sample-intensive article that highlights the strengths of visual programming.

We based part of the article on a programming assignment that James Pirkle recently completed for his client, R.J. Reynolds Inc. We demonstrate how easy it can be to generate an application that will query DB2/2 or, more specifically, access the sample database that ships with DB2/2. We will use Hockware's VisPro/REXX v. 1.1 to generate our DB2/2 sampling application. Additionally, we will show you how to use one of the new DB2/2 support features in the latest version, VisPro/REXX v. 2.0 Gold Edition.

BACKGROUND

VisPro/REXX is tightly integrated with the Workplace Shell. If you're familiar with the Workplace Shell, the development environment will be highly intuitive. Direct manipulation (**Drag&drop**) is used extensively in the development of applications. Additionally, as the name

implies, the product incorporates the OS/2 REXX programming facility. However, as you will see in our sample application, you do not have to be a REXX guru to use the product. Table 1 provides definitions of the terms that we use throughout the article.

SAMPLE APPLICATION

Our sample application, called DB2/2 Sampler, illustrates the use of various OS/2 GUI controls: the notebook, listbox, entry field, push and radio buttons, and valuesets. You can obtain the source code for DB2/2 Sampler from the electronic sources listed in the references box on page 88. We will cover the various steps we followed to create DB2/2 Sampler using VisPro/REXX, so you can reference them.

CREATING A PROJECT

Our first task is to create a new project and forms for all of the DB2/2 Sampler windows. The project will encapsulate all the forms required by the sample application. Figure 1 contains these steps.

INITIALIZING AND ADDING NOTEBOOK PAGES

Notebook pages in Vispro/REXX are really just another form that you add to the form you have designated as a



Event	Occurrence that prompts behavior
Behavior	Function that a form or object performs
Object	A graphical representation of a control or controls
Project	A logical unit of operation such as the sample application
Form	Environment in which the application is designed
Canvas	Area in Layout view in which you add objects to the form

Table 1. Terms used throughout the article

- From the VisPro/REXX PROJECTS folder, drag and drop the Projectfolder template, which will cause a new project to be created.
- Direct Edit the new project folder and change the name to Demo. Open the Direct Edit window using a combination of the <Alt> key and mouse button 1. Close the window by clicking outside of the window.
- Open the Demo folder by double clicking on it with mouse button 1.
- Create the notebook, the two notebook pages, and the DB2/2 Query form.
 - i) Drag&drop the Form template within the Demo folder. This will create a new form. Repeat this three more times.
 - ii) Change the names of the new forms to Notebook, Colors, Fonts, and ShowDept.

Figure 1. Creating a new project

- Open the Fonts form, which will start VisPro/REXX.
- Choose the Open -> Settings menu item from the context menu on the canvas (click with mouse button 2 on the canvas to bring up the context menu).
- Select the Notebook page radio button and close the settings dialogue.
- Drag&drop a radio button object to the canvas.
- Change the text of the radio button to Courier 12pt. Repeat this step for the second Helv 18pt radio button.
- Double click on the top radio button. This will bring up the settings notebook for the radio button.
- Select group and close the dialogue window. Both radio buttons will be added to the same group if the group style is added to only one of them. As a result, only one of the radio buttons can be selected at a time.
- Select the Event tree view menu item from the View menu pulldown.
- From the context menu for the Courier radio button, select the When -> Clicked/selected menu item.
- Open the context menu on the edit region (window on the right). From the context menu, select Add -> Window management -> Notify source window, as illustrated in Figure 3. This will generate the following line:
`CALL VpNotify window, 'message'`
- Change message to FONT 12.Courier. This call will cause a text message to be sent to the parent of this window, which will be the notebook.
- Repeat the prior two steps for the FONT 18.Helv radio button.
- Save the Event tree view.
- From the Form menu pulldown, select Create Link, which copies the REXX command for loading this page into the clipboard.
- Close the Fonts form.

Figure 2. Initializing Fonts page

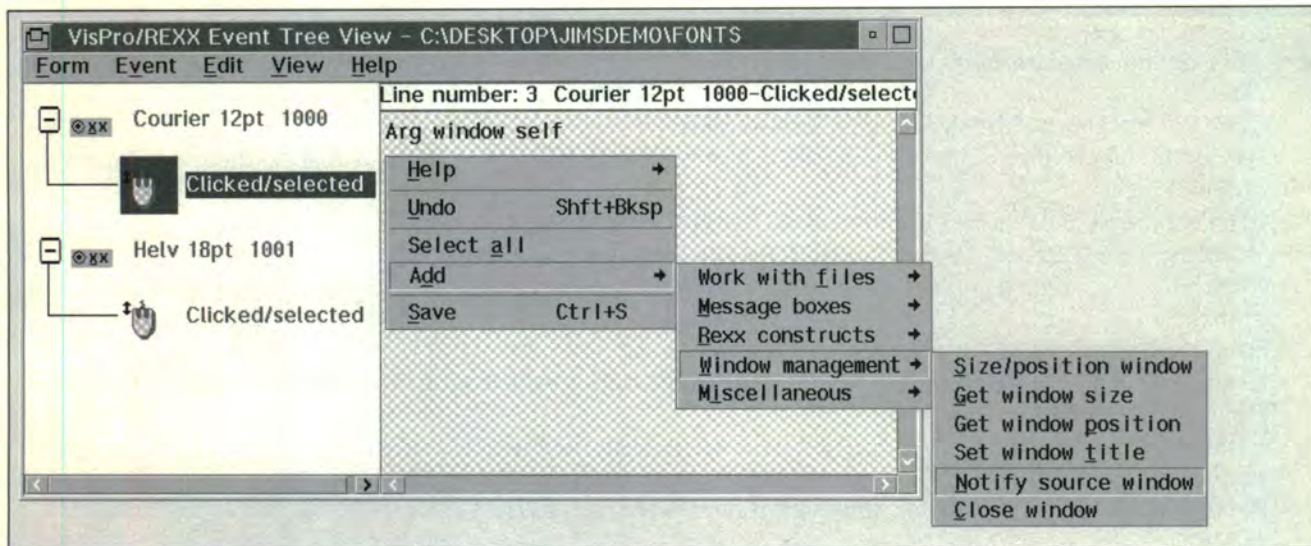


Figure 3. Source window notification step

- Open the Notebook form.
- Drag&drop a notebook to the canvas.
- Select the When -> Opened... submenu from the Form menu.
- From Edit menu on the Event window select Paste, which will put the REXX command that was copied into the clipboard, as shown in Figure 2, in the Event window. This statement will open the Font form when we test DB2/2 Sampler.
- Drag&drop the notebook from the canvas to the Event window. This will bring up the Create Link dialogue.
- Select the Add notebook page item, and change the word title to Fonts. This statement will add the Fonts form to the notebook as a page.
- Save and close the Event window and the Notebook form.

Figure 4. Adding Fonts page

notebook. We will show you how to initialize and add a page to a notebook in this section.

The Fonts notebook form is used to represent a notebook page that will allow selection of two different fonts in DB2/2 Sampler. Radio button controls will be placed on the page to allow selection. Figure 2 lists the steps required to initialize the Fonts page.

Once the Fonts page is initialized, it is ready to be added to the Notebook form. Figure 4 lists the steps required to add the Fonts page.

The Colors notebook form is used to represent a notebook page that will allow selection of eight

different background colors in DB2/2 Sampler. A valueset control will be placed on the page to allow selection. Figure 5 lists the steps required to initialize the Colors page.

Once the Colors page is initialized, it is ready to be added to the Notebook form. Figure 6 lists the steps required to add the Colors page.

DISPLAYING DATA FROM SAMPLE DB2/2 DATABASE

The ShowDept form is used to display data in the sample DB2/2 database that ships with DB2/2. Text fields are created on the form to display

data. Figure 7 lists the steps required to initialize the form.

THE MAIN FORM

The Main form is the form initially displayed when a VisPro/REXX application is invoked. Besides initializing the form, REXX code must be added in order to register the SQL dynamic link libraries (DLLs) and start using the sample DB2/2 database.

Due to space constraints and the lengthy steps required to initialize the Main form, we provide this information in a README file that is part of the source code file that you can download. Specific



- After opening up the **Colors** form and changing it to a notebook page, drag&drop a valueset object to the canvas.
- Open the **Settings** notebook for the valueset control.
 - i) Change the number of horizontal cells to 2 and the number of vertical cells to 4, so there will be a total of eight cells.
 - ii) Select the RGB color radio button, and close the **Settings** view.
- Select the **When** -> **Opened...** submenu from the **Form** menu.
- Drag&drop the value set object from the canvas to the **Event** window and the **Create Link** listbox will appear.
- From the listbox, select **Set cell value**.
- Highlight the generated REXX statement and copy it to the clipboard by selecting the **Copy** menu item from the **Edit** menu.
- Paste from the clipboard seven of these statements into the **Event** window, by selecting the **Paste** menu item from the **Edit** menu.
- Replace the respective variable value with the following: 0, 16777215, 255, 16711680, 65280, 16776960, 65535, 16711935.
- Next replace the row, column variables with 1,1 2,1 3,1 4,1 1,2 2,2 3,2 4,2.
- Save and close the **Event** window.
- Open the **Event** tree view.
- Select the **When** -> **Clicked/selected** option in the context menu.
- Drag&drop the valueset over to the edit region.
- Select **Get item value** from the listbox.
- Open the context menu on the edit region and select **Add** -> **Window management** -> **Notify source window**. This will generate the following line:

```
CALL VpNotify window, 'message'
```

Change the message to **COLOR** value. This call will cause a text message to be sent to the parent of this window, which will be the notebook.
- Save the **Event Tree View**.
- Select **Create Link** from the **Form** menu.
- Close the **Colors** form.

Figure 5. Initializing *Colors* page

- Open the **Notebook** form.
- Select the **When** -> **Opened** menu item.
- From **Edit** menu on the **Event** window select **Paste**, which will put the REXX command that was copied into the clipboard, as shown in Figure 4, in the **Event** window. This statement will open the **Colors** form.
- Drag&drop the notebook from the canvas to the **Event** window.
- Select the **Add notebook page** item and change the word **title** to **Colors**. This statement will add the **Colors** form to the notebook as a page.
- Save and close the **Event** window.
- Select the **When** -> **Secondary notify...** menu item from the **Form** menu.
- Add the word **Parse** before the word **Arg** in the **Event** window.
- Open the context menu and select **Add** -> **Window management** -> **Notify source window** and change message in the resultant REXX statement to **message**.
- Save and close the **Event** window.
- Save and close the **Notebook** form.

Figure 6. Adding *Colors* page



- Open the ShowDept form.
- Drag&drop a push button to the form and edit the text to Ok.
- Drag&drop two text fields to the form, one for a label and the other to display the department information. Open the settings notebook for each and set the Symbol field to TEXT_LBL for the first and TEXT_DEPT for the other.
- Create a When Opened event for the form.
 - i) Select When -> Opened menu item from the Form menu.
 - ii) Open the context menu for the Event window and select Add -> Window management -> Set window title. Change the value variable to Department.
 - iii) Drag&drop the TEXT_LBL text field from the layout view to the Event window. When the Create Link listbox appears, select the Set item value item. Change the value variable to:
 'Department Mgr Division Location'.
 - iv) Append the following lines of REXX code:

```
errmsg = SELDEPT(window topic 'TEXT_DEPT')
      if errmsg \= '' then
        signal SHOW_ERROR
      RETURN 0
      SHOW_ERROR:
```
 - v) Open the context menu for the Event window and select the Add -> Message boxes -> Plain menu item. Change title to Demo Error, and change message to errmsg.
 - vi) Open the context menu for the Event window and select the Add -> Window management -> Close window menu item.
 - vii) Save and close the Event window.
- Open the Event Tree View.
 - i) Open the context menu on the Ok pushbutton, and select the When -> Clicked/selected menu item.
 - ii) Open the context menu on the edit region and select the Add -> Window management -> Close window menu item.
 - iii) Save and close the Event Tree View.
- Save and close the ShowDept form.

Figure 7. Initializing ShowDept form

information on the file is in the reference box on page 88.

SUBROUTINES PROCEDURES (SUBPROCS)

SubProcs are segments of REXX code that can be shared by events within a VisPro/REXX application.

Additionally, they are created with the editor of your choice, the only stipulation being that they reside in the XXX\SubProcs subdirectory, where XXX is the name of your project.

Three SubProcs are defined for the DB2/2 Sampler application, and they provide the following services:

- SELSTAFF formats and adds staff information to the listbox on the

Main form.

- SELDEPT formats and displays selected department information.
- OPENDB registers SQL DLLs and connects to the DB2/2 database.

The source for the three SubProcs can be obtained from the electronic sources listed in the references box on page 88.

TESTING DB2/2 SAMPLER

Finally, we are ready to test DB2/2 Sampler. To test the application, select the Test menu item from the Form menu on the Main form.

If things didn't turn out the way you planned them, make modifications or refer to the sample source code.

In Figure 8, we illustrate a test session. In this illustration, the Open Settings notebook has been opened.

BUILDING AN EXECUTABLE PROGRAM

If everything works as you desire, you can build an executable program by selecting the Build menu item from the Form menu on the Main form.

NEW DB2/2 SUPPORT IN VISPRO/REXX V. 2.0

As you can see, a great deal of our time is spent adding REXX code to perform the queries to the sample DB2/2 database. If this code were generated for us,

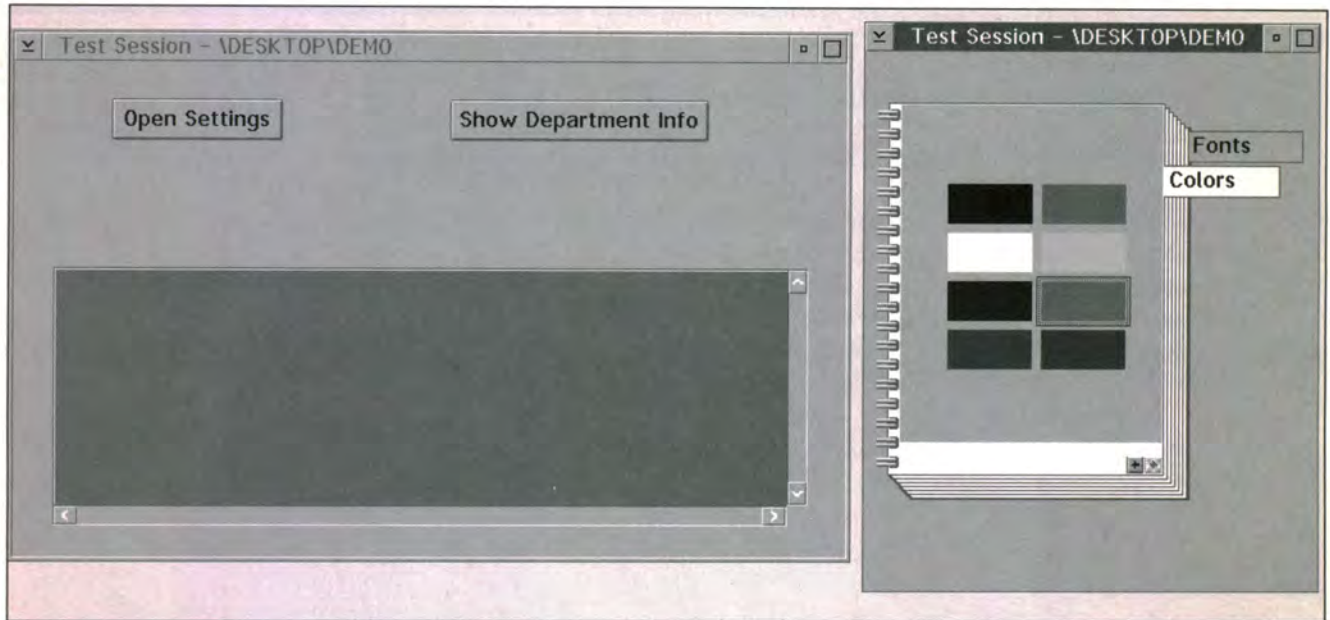


Figure 8. Testing DB2/2 Sampler

- Create a diagram by dragging one off of the VisPro/REXX database diagram template.
- Open the diagram.
- From the DataBase menu, select the Reverse Engineer menu item.
- Select the SAMPLE database from the listbox, click on Find Qualifiers, select USERID from the listbox, and click on OK. The sample database window appears as shown in Figure 10. Observe how the diagrammer has identified the referential constraint relationship between the STAFF and ORG tables.
- Highlight the ORG table object. From its context menu, select Selected -> Open Settings. When the Open Settings notebook appears, select the Attributes tab and the notebook page shown in Figure 11 will appear.

Figure 9. Reverse engineer

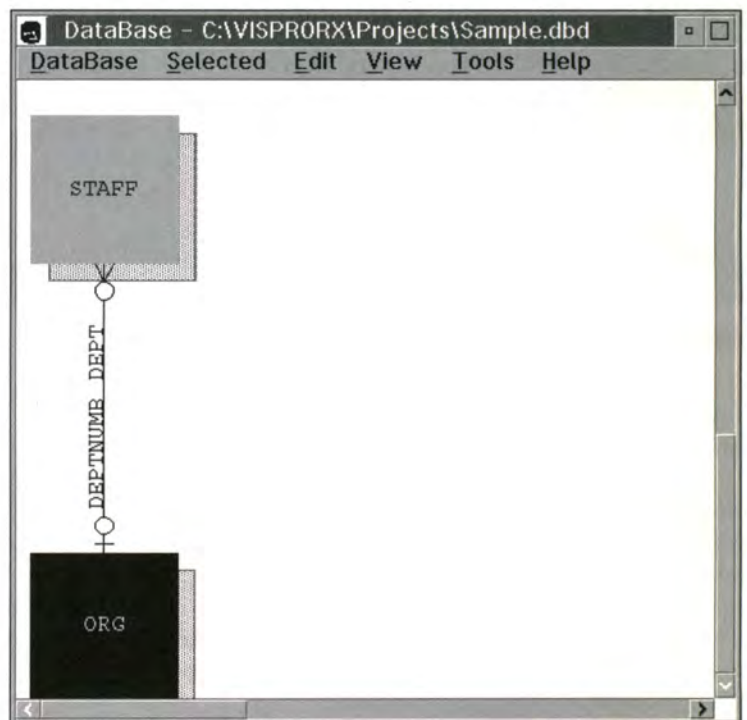


Figure 10. Referential constraint relationship

not only would we immediately become more productive, we would reduce the size of the maintainable code base, which is a blessing to those whose task is program maintenance.

The database support fea-

tures that are in VisPro/REXX v. 2.0 Gold Edition are a logical extension to the v. 1.1 product. If you're comfortable with the work we've done here, the following information will be easy for you to digest.

PREPARING THE SAMPLE DATABASE

To generate a form to access a DB2/2 table, the reverse engineering tool requires the table to have a primary key. To add primary keys to the ORG and STAFF tables, you can enter the following commands at

Column name	Type	Key	Null	Text
DEPTNAME	VCHAR N	Y	Y	Y
DEPTNUMB	SINT	Y	N	N/A
DIVISION	VCHAR N	Y	Y	Y
LOCATION	VCHAR N	Y	Y	Y
MANAGER	SINT	N	Y	N/A

Figure 11. ORG table attributes

DB2/2's command line interface (we also added a foreign key to the STAFF table to illustrate VisPro/REXX's Database Diagrammer):

1. STARTDBM
2. LOGON userid /P:password
3. DBM CONNECT TO SAMPLE
4. DBM ALTER TABLE ORG PRIMARY KEY (DEPTNUMB)
5. DBM ALTER TABLE STAFF PRIMARY KEY (ID) FOREIGN KEY FOR_DEPT (DEPT) REFERENCES ORG ON DELETE RESTRICT

REVERSE ENGINEERING THE SAMPLE DATABASE

Since the sample database already exists, we can take advantage of one of the new features that will allow us to reverse engineer the

sample database. Reverse engineering will display the structure of the database in a graphical format. The steps required to reverse engineer are listed in Figure 9.

THE MAIN FORM REVISITED

In the previous sections that dealt with the Main form and SubProcs, many steps were taken to ensure that all the REXX code was in place to query the sample database. The steps illustrated in Figure 12 replace many of the steps required to initialize the Main form and all of the REXX code entered for the three SubProcs, which is not required in v. 2.0. As you will see, the time and effort required to create applica-

tions by reverse engineering existing databases and Drag&drop is considerably less than what is required to build similar applications by traditional programming techniques.

The steps in the Main form that are not replaced involve adding the Open Settings pushbutton and creating a link to the Notebook form, so it will be opened when a user clicks on Open Settings. Also, the links required to change the Font and Color will need to be implemented based upon the new DB2/2 support.

CONCLUSION

Visual programming encourages rapid prototyping and application development. When compared to a conventional command prompt development environment, the time savings can be measured in hours or even days. However, the pace does slow when you are required to enter code to add support that is not present in the visual programming tool. As the state of these tools advance, so does their functionality.

In the section, *New DB2/2 Support in VisPro/REXX v. 2.0*, we looked at a recent enhancement to the visual programming environment. VisPro/REXX not only saved time, but wear and tear on the keyboard. Unfortunately, we could only cover the reverse engineering portion of the Visual Database Designer enhancement, which can also be used to design DB2/2 databases from scratch.

- Open the Main form.
- Drag&drop the ORG table object onto the Main form. The Create Database Link dialogue will appear, as shown in Figure 13. Use all of the columns and action buttons, which are already selected by default, by clicking on the OK button.
- Save the Main form.
- Test the demo application by selecting the Test menu item from the Form menu, and the test dialogue as shown in Figure 14, will appear. Note the support that allows you to Add, Delete, Change, and Search rows. In the DB2/2 Sampler application, additional code is required to perform these tasks, whereas a simple checkbox selection generates the code for us in v. 2.0 release.

Figure 12. Revisiting the Main form

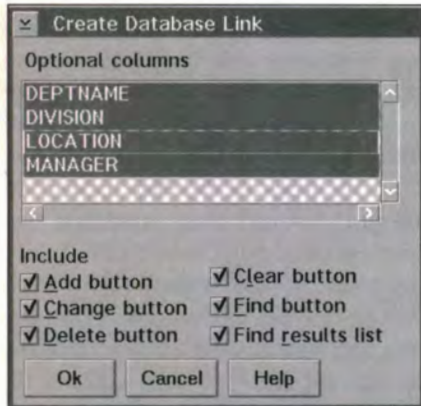


Figure 13. Create database link dialogue

Mark A. Bengé, IBM Programming System Laboratory, Cary, N.C., is a staff programmer who joined IBM in 1989 and has worked on various CUA '91 controls for OS/2 2.x and CUA Controls Library/2. He works in IBM user interface class library development, where he was involved in the implementation of C++ classes for direct manipulation for the C Set++ 2.1 product. Bengé has a B.S. in computer science from Western Carolina University.

John Boezeman, IBM Software Solutions Laboratory, Cary, N.C., is a senior associate programmer who joined IBM in 1990 and has since worked on various C++ projects. He now works in IBM class library user interface development, where he is involved in the implementation of C++ classes for multimedia. Boezeman received a B.S. in computer science from Indiana University.

James Pirkle, Jr., J. Pirkle Inc., Pfafftown, N.C., is an independent consultant with over 10 years of experience writing software for real-time systems. His most recent work includes development of a distributed inventory system using FM hand-held terminals for data acquisition and distributed databases using DB2/2 and MVS host-based databases.

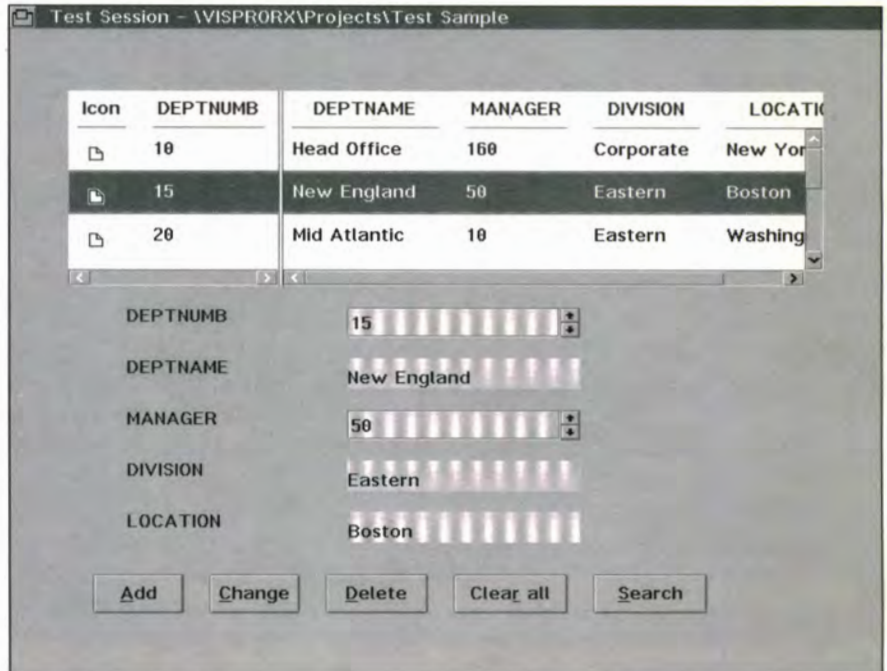


Figure 14. Testing DB2/2 Sampler using new DB2/2 support

REFERENCES

- VisPro/REXX User Manual
- Database 2 OS/2 Guide (IBM Doc. S62G-3663-00)
- Database 2 OS/2 SQL Reference (IBM Doc. S62G-3665-00)
- Database 2 OS/2 Programming Reference (IBM Doc. S62G-3666-00)
- Database 2 OS/2 Programming Guide (IBM Doc. S62G-3667-00)
- Procedures Language 2/ REXX Reference (IBM Doc. S10G-6268-00)

The source code can be downloaded from several electronic sources:

- In the U.S., call the IBM PCC BBS at (919) 517-0001. The source code for this article is in File Area 11 under the name VPRDB2.EXE.
- On CompuServe, the source code is in LIB13 of the OS2DF2 forum, under the name VPRDB2.EXE.
- If you're an IBM TALKLINK customer, the source can be downloaded from the OS2BBS.
- If you have a VM account on an IBM node, get the source code with the command REQUEST VPRDB2 PACKAGE FROM BANZAI AT CARVM3.



The staff of OS/2 Developer put together this special section to guide you through the various LAN development tools available for OS/2. The products described in this guide are based on surveys sent to vendors; they are provided as a service to you and are free to vendors. These listings do not represent an endorsement by OS/2 Developer. Although every effort has been made to ensure accuracy, we do not assume any responsibility for error or omission in this section. Please contact the companies directly for more information.

LAN Development Tools Buyer's Guide

**ANDERSEN CONSULTING—
FOUNDATION ORGANIZATION**

Circle No. 101

FOUNDATION for Cooperative Processing 2.0 provides organizations with the methodology, application development tools, and enablement services for building high-end and enterprise-wide client/server systems. FCP contains FOUNDATION Design, FOUNDATION Production, and FOUNDATION methods, all of which are integrated via the repository and are supported by enablement services. Price: Please contact vendor.

Andersen Consulting—Foundation Organization, 69 W. Washington, Chicago, Ill. 60602, (800) 458-8851 or (312) 507-5161, fax (312) 507-2548.

CAPSTONE SOFTWARE INC.

Circle No. 102

SpaceMap 1.0 for OS/2 is a native 32-bit PM application that analyzes disk space usage at the directory and file level. It summarizes space used by subdirectories and provides file-search, management, and archival facilities. SpaceMap also solves insufficient space problems. It measures LAN application space usage and helps you manage overcrowded LAN drives. Price: \$49.95.

Capstone Software Inc., P.O. Box 416, Carmel, Ind. 46032, (800) 500-2244 or (317) 848-2414, fax (317) 582-0454.

CFSOFTWARE INC.

Circle No. 103

pcMAINFRAME 4.3 is a data transfer software package for PCs and mainframes. It selects, extracts, reformats, and downloads or uploads data to and from PCs and mainframes into existing applications. It offers four levels of security and many selection capabilities. Dialog, a built-in scripting facility, allows users to automate transfers. pcMAINFRAME allows authorized CICS network users to access the data they need. PC requirements: OS/2 1.2 or 3.0. Price: \$9,000+.

Across the Boards 3.11 is an application program interface to communications devices for the IBM PC. It aids you in developing applications that are cooperative between mainframes and PCs. You can develop applications with the following levels

of Across the Boards: DIALOG, a scripting facility/terminal emulator; XAPI, a universal 3270 API; and APPX, a peer-to-peer communications facility. Each level may be purchased separately. Price: \$7,000+.

cfSOFTWARE Inc., 2454 E. Dempster, Ste. 201, Des Plaines, Ill. 60016, (800) 366-8756 or (708) 824-7180, fax (708) 824-0930.

CLIENT SERVER

NETWORKING INC.

Circle No. 104

WATCHIT 1.0 automates collection of LAN SERVER 3.0 capacity and performance data. It allows you to analyze mean response times, bytes transferred, print jobs, buffer allocations, as well as users logged on and resources shared. Charts and graphs help you to make decisions to improve server performance. Certified "Ready! for LAN Server" and "Ready! for OS/2." Price: \$299 per server.

CONDUIT 2.0 is a network-independent LAN file distribution system for installation, upgrade, and version control on any number of networked DOS or OS/2 clients. It creates directories, remotely executes programs, and maintains inventory data. Collected profiles allow you to reconstruct system files. Certified "Ready! for LAN Server." Price: \$299 per server.

Client Server Networking Inc., P.O. Box 370111, West Hartford, Conn. 06137-0111, (203) 233-2951, fax (by arrangement) (203) 233-2951.

**COMPUTER ASSOCIATES
INTERNATIONAL INC.**

Circle No. 105

CA-Realizer 2.0 is an OS/2-based visual application development tool. Using BASIC as its core language, CA-Realizer provides a development environment for client/server, LAN, departmental, or single user applications. You may distribute applications built with CA-Realizer free of royalty or license fees. CA-Realizer is compatible with OS/2 2.1 or higher. Price: \$99, until March 31, 1994; \$247, after March 31, 1994.

Computer Associates International Inc., One

Computer Associates Plaza, Islandia, N.Y. 11788-7000, (800) 225-5224 or (516) 225-5224, fax (516) 342-4864.

DEVELOPMENT

TECHNOLOGIES INC.

Circle No. 106

DeskMan/2 1.2 helps you to manage Workplace Shell desktops and objects locally and remotely. It also allows you to authorize access to individual DeskMan/2 features on a user-by-user basis. Deskman/2's features include security and window management, GUI and API to manipulate WPS objects, and configuration and desktop snapshot facility. It is network-enabled, including CID and LAD/2. Price: \$79.95 (street price varies).

Development Technologies Inc., 308 Springwood Rd., Forest Acres, S.C. 29206-2113, (803) 790-9230, fax (803) 738-0218.

EASEL CORP.

Circle No. 107

ENFIN 2.9 is an object-oriented development toolset for Windows, OS/2, and UNIX. It delivers reusable applications that can be deployed across a multiplatform network. Price: Please call.

Easel Corp., 25 Corporate Dr., Burlington, Mass. 01803, (800) 759-5880 or (617) 221-2100, fax (617) 221-6899.

EICON

TECHNOLOGY CORP.

Circle No. 108

APPC Application Subsystem 1.00 for OS/2 supports off-the-shelf applications written to the IBM Communications Manager/2 APPC specifications. It works with the Eicon Technology SNA LAN Gateways (DOS, OS/2, and NetWare servers) and the SNA PC Gateway for OS/2. It provides an interface for session management and administration and allows user programs to establish peer-to-peer sessions with other PCs, midrange computers, and mainframe computers. Price: \$195+, depending upon numbers of users; \$3,750 for LAN Gateway Pack.

SNA Function Management Developer's Toolkit 1.00, through an Eicon Technology SNA Gateway, allows PC or PS/2-based applications to set up Application-Application (LU-LU) simultaneous sessions with an SNA host, including IBM S/370, System/3X and AS/400 computers. It provides an interface for the SNA protocol stack, using a library of C function calls. Price: \$1,000.

EiconAPPC Developer's Toolkit 2.00 for OS/2 facilitates the development of OS/2-based communications applications based on the IBM LU 6.2 protocol. Using a library of C function calls to provide a high-level interface to the SNA protocol stack, it allows developers to design programs that interact on a peer-to-peer basis with other local or remote applications on an SNA/APPN

network. Other applications can be running on other personal computers as well as any minicomputers or mainframes supporting APPC. Price: \$1,000.

Eicon Technology Corp., 2196 32nd Ave., Lachine, Quebec H8T3H7, Canada, (514) 631-2592, fax (514) 631-3092.

GATEWAY

SYSTEMS CORP.

Circle No. 109

SYNERGIST Application Development System 3.6 is a client/server application development tool that integrates PCs with IBM, HP, and DEC hosts. It helps you to develop on-line, transaction-based applications that execute on the PC and access existing databases, PC databases, or both. You can use SYNERGIST to develop order management, distributed processing, and sales force automation applications. Price: Please call.

Gateway Systems Corp., 4660 S. Hagadorn Rd., Ste. 110, East Lansing, Mich. 48823, (800) 333-9366 or (517) 337-8960, fax (517) 337-2868.

GILMORE SYSTEMS

Circle No. 110

Magnum BBS 7.00c is a customizable electronic BBS for OS/2 with file transfer, electronic e-mail, CD-ROM support, offline messaging, and RJE (background) program execution. You can use it as a modemless LAN-based BBS or LAN-based e-mail system. It supports OS/2 and DOS client workstations. Price: \$75-1,000+.

Gilmore Systems, 679 Switzer Ln., Thousand Oaks, Calif. 91360, (805) 379-3210, fax (805) 379-1341.

GLENCO

ENGINEERING INC.

Circle No. 111

Hardlock Master Kit-OS/2 consists of a combination of software and a small hardware key or internal board that attaches to a PC. It features an algorithmic response ASIC with optional RAM/ROM for field programmability. Systems available for parallel and serial ports, DOS, UNIX, Xenix, OS/2, Windows, CD-ROM, single-user and network configurations. Price: \$225.

Glenco Engineering Inc., 270 Lexington Dr., Buffalo Grove, Ill. 60089, (800) 562-2543 or (708) 808-0300, fax (708) 808-0313.

INTELLIGENT

ENVIRONMENTS

Circle No. 112

AM (Applications Manager) 4.0 is a programming environment for building and maintaining client/server applications. It enables you to develop 32-bit, multitasking, object-oriented applications. Other features include visual programming facilities. You can divide development among team members on a LAN through reusable components accessed from shared libraries. Price: \$8,400 (includes one year of support and upgrades).

AM (Applications Manager)/SQL

Workbench 2.1 is a visual development tool for designing, compiling, and testing static SQL transactions in an SAA environment. It enables client/server applications that otherwise would require dynamic SQL to execute multi-threaded static SQL. Price: \$8,400 (includes one year of support and upgrades).

AM (Applications Manager) CP/Work-

bench 2.0 is a graphical tool for creating and maintaining APPC transactions for "cooperative processing" client/server applications. It also aids in defining conversations and building and parsing communications buffers. Because the compiled definitions are maintained in a repository, subsequent modifications do not impact the application. Price: \$8,400 (includes one year of support and upgrades).

Intelligent Environments, Two Highwood Dr., Tewksbury, Mass. 08176, (800) 669-2797 or (508) 640-1080, fax (508) 640-1090.

IPSWITCH INC.

Circle No. 113

Vantage/IP 1.0 is TCP/IP software for OS/2 with a set of OS/2 and DOS applications. You can use Vantage/IP's set of APIs to run a wide variety of third party TCP/IP applications. Vantage/IP transfers data at speeds exceeding 500k bytes per second. Price: \$390.

Ipswitch Inc., 333 North Avenue, Wakefield, Mass. 01880, (617) 246-1150, fax (617) 245-2975.

LANE

TELECOMMUNICATIONS

Circle No. 114

Passport 3000 4.0 is an OS/2 software application that allows an Intel-based PC, connected to a LAN, to act as a communications server. The Passport package supports up to 32 gateways including: e-mail, facsimile, telex, X.25, direct distance dialing, and leased lines, on a single system. Passport processes messages between gateways as a store and forward message handling system. It provides a set of administrative utilities both to monitor and manage the message traffic. You can generate detailed reports by user, communication line, and department. Price: \$3,500-21,000+.

Lane Telecommunications Inc., 5 Marineview Plaza, Ste. 210, Hoboken, N.J. 07030, (201) 798-0006, fax (201) 798-0045.

MARKHAM

COMPUTER CORP.

Circle No. 115

NetOpPM 4.21 is a remote-control product for networked PCs. It allows the control of, screen view, and file transfer between any OS/2, DOS, or Windows PC in the LAN or WAN. It will remote-control servers as well as workstations. It supports NetBIOS and

Novell IPX/SPX. Price: \$1,495 for a master and up to 50 remotes; \$795, each additional master.

Markham Computer Corp., One S. Ocean Blvd., Ste. 207, Boca Raton, Fla. 33432, (800) 262-7542 or (407) 394-3994, fax (407) 394-3844.

MERGENT INTERNATIONAL INC. *Circle No. 116*

PC/DACS 1.03 for OS/2 includes features like ID and password-based protection for OS/2 2.x platform; session time-out; boot protection (dual-boot combined with PC/DACS for DOS); several levels of encryption; and multi-user set up. It is transparent to users and protects information without impeding performance. You can accomplish enterprise-wide administration through Site/DACS. You can customize PC/DACS per organizational security requirements, then deploy and install it across thousands of workstations. Price: \$189, unit; \$4,500 for site license of 25 units (includes Site/DACS).

Mergent International Inc., 70 Inwood Rd., Rocky Hill, Conn. 06067, (203) 257-4223, fax (203) 257-4245.

MICROFORMATIC/AMERICAN TELEREP INC. *Circle No. 117*

Fax/PM LAN 3.0 is a client/server that uses Distributed SOM. It supports NetBios, Novell, and TCP/IP. Fax/PM LAN also supports DOS, Windows, and OS/2 clients. Its Restart checkpoint allows you to resume transmission from point of failure. FaxPM supports multiple user-definable channels and multiple phone lines in send and receive mode. Features include workgroup and private phone books for e-mail; fax routing from DID and DTMF; and CID installation. REXX and SOM API's are included with program.

Microformatic, 2 rue Navoiseau, Montreuil-Sous-Bois, F-93100 France, 33(1)48-70-19-00, fax 33(1)48-70-27-29. American TeleRep Inc., 610 Niederwerfer, South Windsor, Conn. 06074, (203) 644-1708, fax (203) 648-9587.

MICROWORK INC. *Circle No. 118*

Job Scheduling Server 1.0 is an OS/2 32-bit PM version of the Job Scheduler for IBM and Novell LANs. It schedules OS/2, Windows, and DOS jobs. It allows unattended job scheduling on specified or auto-selected nodes and conditional job scheduling based on other jobs return codes or presence files. An on-line LOG keeps screens and messages captured during job execution. Job Scheduling Server controls the number of jobs concurrently executing on the same node. Features include job priority, calendar, and security. Ships on May 15, 1994. Price: Please call.

Microwork Inc., 47 W. St. Andrews Ln., Deerfield, Ill. 60015, (708) 940-8979, fax (708) 940-8979.

NETTECH SYSTEMS INC. *Circle No. 119*

Nettech Rfgate 1.3 is a host gateway to wireless digital radio networks, including Ardis and RAM Mobile Data. The API allows host applications to run on OS/2 or application stubs to route data to or from other hosts from OS/2 gateway. Price: \$695 run time, \$4,695 for SDK.

Nettech RfLink 2.0 is a transport designed for wireless digital radio networks. It runs on OS/2 under Nettech Rfgate and can simultaneously send and receive multiple messages or files from multiple destinations mobile units. Requires Nettech RfLink for DOS or Windows on mobile side. Price: \$895 run time; \$2,695 SDK.

Nettech Systems Inc., 324 Wall St., Princeton, N.J. 08540, (609) 683-0100, fax (609) 683-5019.

OPEN SOFTWARE ASSOCIATES *Circle No. 120*

OpenUI 2.13 is a cross-GUI development system for client/server and stand alone applications. It provides portability across graphical user interfaces and also manages the messaging between client and server systems. OpenUI provides scalability for enterprise-wide client/server applications. It supports C, C++, COBOL, Ada, Pascal, and fourth-generation languages; a wide variety of client and server platforms; and standard version control and testing tools. Price: Varies by platform and quantity.

Open Software Associates, 20 Trafalgar Sq., Ste. 414, Nashua, N.H. 03053, (603) 886-4330, fax (603) 598-6877 or 1250 Oakmead Pkwy., Ste. 210, Sunnyvale, Calif. 94088-3599, (408) 730-2626, fax (408) 730-2619.

PARALLEL PCS INC. *Circle No. 121*

Synetics SDK 2.0 for OS/2 helps you to develop Synetics Aware applications that enable you to locate and apply idle processing resources to your processing needs. By using the idle resource found on LANs for parallel processing, you benefit from shorter execution times. The Synetics SDK provides the source code, libraries, and documentation needed for you to build Synetics Aware applications. Price: \$2,200.

Parallel PCs Inc., 1404 Durwood Dr., Reading, Pa. 19609, (610) 670-1710.

PEERLOGIC *Circle No. 122*

PIPES Platform Developer's Kit 3.5 is message-oriented middleware that provides an infrastructure for distributed computing. With it, you can build distributed applications that run over diverse operating systems and network protocols. PIPES Platform resides between the application and the network, providing communications services such as

guaranteed message delivery, distributed naming services, resource location transparency, dynamic rerouting and error recovery, and automatic context bridging between different network protocols. Price: \$495; call for additional licensing information.

PeerLogic, 555 DeHaro St., San Francisco, Calif. 94107-2348, (800) 873-7927 or (415) 626-4545, fax (415) 626-4710.

PINNACLE TECHNOLOGY INC. *Circle No. 123*

Desktop Observatory 3.0 allows you to configure desktops remotely and standardize OS/2 clients regardless of network operating system. Features include mobile desktop and password security when you log on any workstation. You can protect the Observatory or use the DCE security service interface. Desktop Observatory helps you to prevent drags, deletes, copies, renames, moves, shadows, or just hide them. It launches your REXX or C based event. Price: \$149.

Pinnacle Technology Inc., P.O. Box 128, Kirklin, Ind. 46050, (800) 525-1650 or (317) 279-5157, fax (317) 279-5157.

QUADRON SERVICE CORP. *Circle No. 124*

qCF 3.5 is a C-language development and runtime environment for multitasking communications applications using IBM ARTIC coprocessor cards. It includes drivers, loaders, and symbolic analysis debug tools. Application design consists of open/close, read/write concepts. Use qCF to develop BISYNC, ASYNC, SDLC/HDLC, and custom protocols. Price: \$2,499, qCF Developer's Toolkit; \$279, qCF Runtime Support.

qX25 1.8, an open-architecture X.25 implementation, uses IBM ARTIC cards and includes fast select support, communication statistics, packet and frame trace, and multiple lines per cards. All levels of X.25 are a single ARTIC task. You can specify up to 100 virtual circuits per line, and adjust frame, packet, and time-out parameters. Price: \$499, qX25 Developers Toolkit; \$399, qX25 Runtime Support.

qLAPB 1.2 supports link-layer LAP-B protocol. It utilizes a qLAPB task in an IBM ARTIC coprocessor card and contains frame-level procedure code that implements LAP-B. The qLAPB task communicates with a users application through an API supplied with the software. Price: \$699, qLAPB Developers Kit; \$499, qLAPB Runtime Support.

Quadron Service Corp., 209 E. Victoria St., Santa Barbara, Calif. 93101, (805) 966-6424, fax (805) 966-7630.

RIGHTFAX *Circle No. 125*

RightFAX 3.5 is a LAN fax server that supports





LAN Manager, LAN Server, and NetWare as well as OS/2, DOS, and Windows workstations. You can route incoming faxes manually or automatically using direct-inward-dialing, DTMF, or PBX interface. The RightFAX E-Mail Gateway facilitates sending and receiving faxes from cc:Mail, Lotus Notes, and Microsoft Mail. Price: Please call.

RightFAX, 4400 E. Broadway, Ste. 312, Tucson, Ariz. 85711, (602) 327-1357, fax (602) 321-7456.

SAS INSTITUTE INC.

Circle No. 126

SAS System 6.0 is an integrated suite of software products for enterprise-wide information delivery. It provides you with tools to access, analyze, and present data within an applications development environment. The SAS System includes modular components for cooperative and distributed process and a computer performance evaluation tool that allows systems analysts to collect, analyze, and report current usages. Other capabilities include database access, applications development, and client/server computing. Price: Please call.

SAS Institute Inc., SAS Campus Dr., Cary, N.C. 27513, (919) 677-8000, fax (919) 677-8123.

S-CUBED INC.

Circle No. 127

AnyClient 1.1 is an add-on, programmable, middleware layer that extends the capability

of ODBC, IDAPI, DB-LIB, and QE-LIB. With AnyClient installed, you can write client software using visual development tools or DBMS tools. In practice, you use AnyClient to define a "client's view virtual database" mapped to real database objects, thus making everything, even multiple joins, look like a base table. In production, a translator intercepts all messages between clients and the server; remaps views to the appropriate tables; and invokes any stored procedures that may exist. You are free to tune and modify your server. Price: \$2,995 per LAN.

SQL Server to DB2 Rapid Migrator 1.1

enables automated migration of applications from a Sybase/SQL Server environment to a DB/2 and DB2 6000 environment. It allows you to execute stored procedures and trigger code developed with SQL Server in a DB2 environment. It also permits one set of code to support multiple environments, thus preserving investment in existing code and reducing maintenance. DB-Lib transparency software eliminates the need to make changes on the client side when changing server. Price: \$2,995 for development system; \$995 for OS/2 server production system.

Pro2Pro 1.1 is a database design and code generation tool with both forward and reengineering capability. It produces DDL and DML for most SQL-

RDBMS and generates Transact SQL, COBOL, and C. You can map and reuse data and code definitions from existing systems to new requirements. Pro2Pro then generates the remaining database definitions and code that is required to meet the requirements. It supports multiple production platforms from the same set of requirements. Pro2Pro also supports change impact analysis and reporting; features automatic detection and generation of any required data conversion products; and facilitates tuning and denormalization. Price: \$7,995.

Forms2Data 1.1 is an RDBMS design and prototyping tool for client/server systems. Developers gather business requirements starting with how the application looks to the outside world and how information on the "forms" is related to other information, computed, and looked up. They specify the tasks to be performed and who performs them. Forms2Data then uses its "applied normalization" concepts to design the required database structure and generate a functional prototype. The application is then tested using the prototyper and modified as required until the business functionality is achieved. The prototype can be put into production "as is" or transferred to the implementation tool, Pro2Pro. Price: \$2,995.

MEGA POWER FOR OS/2 BACKGROUND APPS

- ▶ Set the priority OS/2 Apps in Full Screen or Window mode including Apps installed as icons. Get information to help prioritize DOS Apps.
- ▶ BIG performance gains in heavy multitasking and/or memory overcommit scenarios or as little as two background Apps via prioritization.
- ▶ Indepth Online Help with Tips, Questions and Answers NEVER BEFORE PUBLISHED Find out how faster machines just "spin" faster and mixing DOS and OS/2 is a disaster.
- ▶ The most powerful Application ever designed for OS/2. It is more effective than multiple processors for Background Apps.
- ▶ We are the steering wheel and gas pedal for OS/2. If you don't have our products—YOU'RE OUT OF CONTROL!
- ▶ "READY FOR OS/2".
- ▶ 32 bit Presentation Manager Application (works on OS/2 2.0 & 2.1). 30-Day money back guarantee! Absolutely required for ALL OS/2 systems. NEW LOWER PRICES.
- ▶ Background Master I/II—32/64 priority levels. \$19/\$29
- ▶ Priority Master I/II—96/128 priority levels. \$39/\$49

ScheduPerformance Inc.

3474 N. University Dr. • Suite 217 • Sunrise, Florida 33351

Voice (305) 486-8299, Fax (305) 486-5018 • M-F 10:00AM to 8:00PM EST



S-Cubed Inc., 1010 Washington Blvd., Stamford, Conn. 06901, (203) 323-0760, fax (203) 323-2007.

SATORI AUTOMATION **Circle No. 128**

ScriptMate 1.0 allows users, system analysts, and programmers to automate routine micro-to-mainframe sessions. It is compatible with most 3270 formulators on the market and comes packaged with versions for DOS, Windows, and OS/2. ScriptMate is teamed with ScheduleMate, a scheduler that allows users to schedule the execution of programs and batch files. Price: \$295, single user; \$995, five-user LAN.

Satori Automation, 770 Old Roswell Pl., Ste. D400, Roswell, Ga. 30076-1649, (404) 640-0904, fax (404) 518-3331.

SERENGETI SYSTEMS INC.

Circle No. 129

3780Link 2.5.0 provides 3780/2780 RJE for OS/2, UNIX, AIX, and MS-DOS. You can use it for EFT, EDI, POS, ACH, claims filing, and other bisync (BSC) applications. It has a menu-driven interface that helps you to install, configure, and operate it. You can use the script language for unattended sessions. You can invoke Script sessions from custom applications. Multiple BSC ports can run on a single system. A variety of internal and external modems are supported. Price: \$795-2,895.

BSCLIB 1.6.5 is a bisync (BSC) API for OS/2, UNIX, AIX, MS-DOS, and Windows, which gives you access to 3780/2780 communications. You can use BSCLIB in RJE, EFT, EDI, POS, ACH, claims filing, and other BSC applications. The API shields programmers from the details of the BSC protocol. Auto-dial, auto-answer, and extensive status codes aid in unattended sessions. BSCLIB can control multiple BSC ports on a single system. A variety of modem options are available. Price: \$595-2,895.

Serengeti Systems Inc., 808 W. 10th St., Austin, Tex. 78701-2042, (800) 634-3122, fax (512) 480-8729.

SOFTTOOLS CORP.

Circle No. 130

CCC (Change and Configuration Control)/Manager 2.1 for OS/2 is targeted at software developers, technical testing staff, quality assurance personnel as well as production control and library managers, working either stand alone or on a network. CCC/Manager is part of Softool's CCC product family, offering life cycle management functionality across 20 platforms including UNIX, IBM, Digital, and the PC. Features include an object-oriented user interface; modeless execution that permits simultaneous operations; central list that can be kept open all the time; and mirroring PC directories in CCC. CCC/Manager also

provides support for concurrent development, which allows users to work simultaneously on the same code, and a data management facility that allows you to unload information into tab-delimited files. CCC/Manager permits incorporation of CCC/Bridge in a distributed MVS/PC environment.

Softools Corp., 340 S. Kellogg Ave., Goleta, CA 93117, (805) 683-5777, fax (508) 683-4105.

SOFTWARE AG

OF NORTH AMERICA

Circle No. 131

ADABAS Server 1.25. Software AG's premier database management systems to the workstation. This implementation highlights ADABAS's well-known strengths of being able to handle both high transaction volumes and complex ad hoc queries, while operating against large volumes of data. Thus, ADABAS DB Server is well-positioned for many types of applications, both strategic and operational. ADABAS is supported across OS/2, Windows NT, UNIX, OpenVMS, DOS/VSE, VM/CMS and MVS operating environments. Price: \$500+.

NATURAL (Application Development) is a fourth-generation environment for users and developers. NATURAL'S interactive capabilities help users design, develop, test, manage, deploy, and maintain applications. Applications are performance-opti-



HAVE THE WORLD AT YOUR FINGERTIPS



THE OS/2 LAN SERVER ADVANTAGE

Connect to the LAN Server advantage with the help of our OS/2 SWAT Team.

Let's face it, connectivity and communication is the key to the world. That's why we've spent the last six years making these fields our specialty.

We offer

- OS/2 LAN Server Installation
- OS/2 LAN Server training
- Advanced LAN Server training
- Full operational support
- Custom fine tuning of your system.

We also offer in-depth training in OS/2 Administration, REXX Programming, Base Programming, OS/2 Presentation Manager, ANSI C, and Database Manager.

Adaptive Research & Design Company

Miami

Call: (305) 899-0700 or Fax (305) 892-8669

TLIBTM Version Control **For DOS, OS/2 and Windows-NT**

• The experts loved TLIB 4:

"...amazingly fast... TLIB is a great system." **PC Tech Journal**
"TLIB has features and power to spare... TLIB is easy to use and the fastest of the reviewed packages." **Computer Language**
"I will not program without it." **Uptime Magazine**

• Now TLIB 5.0 adds:

Automatic branching. Automatic version labeling across branches. User defined **promote** structures, for staged development. Exclusive whole-level **change migration** for customized software. N-way-tree version numbers. Branch and full locking. OS/2 & NT support.

• Plus the features they loved in TLIB 4:

Check-in/out locking. Branching, for parallel development. Keywords. Full binary file support (does not depend upon CRs in the file like other products). Wildcard and list-of-file support; can create lists by scanning source code for includes. Can merge (reconcile) multiple simultaneous changes and undo intermediate revisions. Network and WORM optical disk support. Mainframe-compatible delta generator for Pansophic, ADR, IBM, Sperry formats. Includes integrated PD MAKE by L. Dyer; also integrates with OpusTM MAKE, SlickTM MAKE, others.

MS-DOS **\$139**, OS/2 & NT (with MS-DOS) **\$195** + shipping.
5 station network: MS-DOS \$419, OS/2 \$595. Call for other sizes.



Burton Systems Software

PO Box 4156, Cary, NC 27519 **(919) 233-8128**
FAX: 233-0716



mized for multivendor distributed computing environments. NATURAL's users benefit from easy-to-use graphical user interfaces, comprehensive HELP system, interactive development and testing, and transparent user access to LAN, midrange, and mainframe resources and portability of applications, data and communications within diverse distributed environments.

ENTIRE NET-WORK (Distributed Databases) 2.1 allows applications residing on different platforms and in different geographical locations to appear to users as if data were residing on one platform in one location. Users can access multiple databases. For example, data can be extracted from remote databases, processed within a local session, and returned to the remote database. Users will have a singular view of enterprise data. An intuitive graphical user interface allows network administrators to support and maintain the enterprise network.

Software AG of North America, 11190 Sunrise Valley Dr., Reston, Va. 22091, (800) 423-2227 or (703) 860-5050, fax (703) 391-8360.

**SOFTWARE CORPORATION
OF AMERICA**

PolyPM/2 2.2 for OS/2 allows an OS/2

Circle No. 132

"Teacher" PC to control a DOS, OS/2, or Windows "Pupil" PC. These PCs may be connected via modem, NetBIOS, APPC/APPN, X.25, IPX, or TCP/IP. You can use PolyPM/2 for remote LAN support, training, or customer support. Price: \$295.

TalkThru 2.2 for OS/2 provides terminal emulation, file transfer, scripting and EHLLAPI for network (connected via NetBIOS or TCP/IP) or modem-connected PCs. TalkThru's EHLLAPI support allows PCs to connect to both IBM and non-IBM hosts. Price: \$190.

Software Corporation of America, 100 Prospect St., Stamford, Conn. 06901, (800) 966-7722 or (203) 359-2773, fax (203) 359-3198.

**THE SOFTWARE
LIFELINE INC.**

REMOTE-OS 2.10 Multi-User for OS/2 is ready to use, or developers can take advantage of the toolkit to provide special support for, for example, Data Collection devices, and access OS/2 applications. Access is via RS232. Up to 48 concurrent terminals can share the OS/2 applications on a single OS/2 applications server. Applications like cc:Mail, WordPerfect, or even 3270 emulation can concurrently be run from terminals like a VT100 or even a 3270 MainFrame terminal. Price: \$99+.

The Software Lifeline Inc., Executive Court One, 2295 Corporate Blvd.,

Circle No. 133

Boca Raton, Florida 33431, (407) 994-4466, fax (407) 994-6304.

**STRATEGIC SOLUTIONS
INTERNATIONAL CORP.**

Circle No. 134

Net/Impact 1.0 is an SNA Resource and MVS Application Monitoring and Management System. Net/Impact SNA/Agent runs on an MVS host and can co-exist with NetView or Net/Master. Net/Impact PM/Workstation runs on OS/2 2.x, providing network operations interface. It provides service level objectives, real-time, dynamic network status, and automated or manual VTAM command interface. Price: Please call.

Service Point/32 1.5 is a NetView service point and OS/2 automation development platform. NetView FUNCMD, GENALERT, MSG, and VIEW support within OS/2 REXX. It provides terminal emulation for VT-102, IBM 3101, ANSI 3.164, UIBM 3164, and Glass TTY. Features include data scope, capture/simulation/playback facilities. Requires Communication Manager/2 but does not need ROPS or NetView/PC. It provides communications via RS-232, Named pipes. Price: \$3,000 or higher, depending upon configuration.

Strategic Solutions International Corporation, 1075 Tolland Turnpike, Manchester, Conn. 06040, (203) 649-1900, fax (203) 649-1230.

**Read and Write dBASE files
from your C/C++ programs!**

Full package: \$ 195
REXX only: \$ 89

dbfLIB

Programmer's Library

New!

Version 2 Features:

- ↳ Supports new index formats
- ↳ REXX support included
- ↳ New streamlined functions
- ↳ Royalty free DLLs for Windows, OS/2 and Windows NT
- ↳ File and record locking APIs
- ↳ Handle based calls
- ↳ Memo field & index support
- ↳ Libraries for DOS, Windows, OS/2, and Windows NT

**New!
REXX DLL
Windows NT DLL**



dSOFT Development Inc.
4710 Innsbruck Drive
Houston, TX 77066

Sales (405) 360-3045
Tech Support (713) 537-0318
Visa and MasterCard accepted.

Circle Reader Service Number 47

**Entryfield
picture
masks**

for OS/2

The Impact Formatted Entryfields are a completely new window class which allows the developer to specify a format string when the entryfield is created. Formatting can be used for fields like Phone Number, Dates, Social Security Number, etc.

The Impact Formatted Entryfield dynamically formats itself as the user inputs data.

The Impact Formatted Entryfield support assures uniformity in data entry.

The developer has complete control to create custom fields like;

- Numeric entry only
- Alpha characters only
- Special delimiting characters
- Value ranges
- Currency support and, much more.

Easy to use! Just change the window class in your resource file.

Download free demo of Impact Entryfields from our BBS.
(818) 879-7405

Add Impact to your application



IMPACT SOFTWARE

5889 Kanan Rd.
Suite 330
Agoura Hills
CA. 91301

To order call or write
(800) 676-9390
(818) 879-5592
FAX (818) 879-5593

Circle Reader Service Number 48

SYZYGY**COMMUNICATIONS INC. Circle No. 135**

XAM/OS/2 1.0 provides file transfer between remote DOS and OS/2 workstations, between AS/400s and OS/2 workstations, and between any LAN (for example, OS/2 LAN Server) and any other LAN (for example, NetWare). Price: \$1,500.

Syzygy Communications Inc., 269 Mt. Herman Rd., Ste. 201, Scotts Valley, Calif. 95066, (408) 438-5111, fax (408) 438-5115.

TOKEN**TECHNOLOGY INC. Circle No. 136**

RemoteVision 2.0 extends Token-Ring networks to remote workstations and workgroups over asynchronous modems. LAN applications written to one of the LAN interfaces like NetBIOS, APPC, 802.2, IPX, TCP/IP, or XNS work transparently over remote connections. RemoteVision supports the following configurations: remote-to-LAN, LAN-to-LAN, and remote-to-remote. Price: \$695+, depending upon number of users.

Token Technology Inc., 1265 Montecito Ave., Ste. 101, Mountain View, Calif. 94043 (415) 965-8607, fax (415) 965-8658.

UNIFACE CORP.**Circle No. 137**

UNIFACE 5.2 for OS/2 is a development environment that helps you build client/server

applications that take advantage of the native features of OS/2. Applications built with UNIFACE are open and portable and may be deployed throughout your entire enterprise. In addition to OS/2, UNIFACE is supported on many hardware platforms, operating systems, GUIs, networks, and data servers. UNIFACE for OS/2 is a native 32-bit OS/2 2.x application. UNIFACE applications adopt the native look and feel of Workplace Shell, supporting new controls including the Notebook Settings Object. Price: \$2,495.

Uniface Corp., 1320 Harbor Bay Pkwy., Ste. 100, Alameda, Calif. 94025, (800) 365-3608 or (510) 748-6145, fax (510) 748-6150.

WALL DATA INC.**Circle No. 138**

RUMBA for the Mainframe 3.2 brings IBM mainframe applications to the PC. It supports interfaces and LAN connections, including high-speed and dial-up async SNA connections, and implements APPC for client/server applications. Price: Please call.

RUMBA for NetWare System brings IBM mainframe applications to the PC. It supports NetWare gateways—NetWare for SAA or the Microdyne EXOS (formerly NetWare) SNA Gateway—via IPX/SPX. RUMBA for the Mainframe includes Novell LAN WorkPlace TCP/IP protocol stack. Price: Please call.

Wall Data Inc., 17769 NE 78th Pl., Redmond, Wash. 98052, (800) 222-0861 or (206) 881-5995, fax (206) 861-3175.

XDB SYSTEMS INC.**Circle No. 139**

XDB-Workbench 3.2 is a DB2-compatible development and production environment available for PCs, LANs, and workstations. It consists of a DB2-compatible SQL database engine, a COBOL precompiler, and a suite of development and administrator utilities. The XDB-Workbench enables users to develop, test, and run COBOL programs with embedded SQL on PCs, LANs, and workstations. You can migrate programs to the mainframe. XDB-Workbench can be used stand-alone, as a client accessing any XDB-Server on a LAN, or connected to the mainframe DB2 via XDB-Link. Price: Please call.

XDB-SQL RDBMS 3.2 is a database engine, running under Microsoft Windows that provides 100% of IBM's mainframe DB2 SQL on PCs, LANs, and workstations. XDB-SQL RDBMS includes XDB's SQL DLL Engine for Windows, an ODBC driver, a DDE interface, and a redirector for LANs. XDB's DLL SQL Engine for Windows is multiuser and capable of supporting multiple concurrent Windows applications. Applications written in C or interfacing to ODBC or DDE can access XDB-SQL RDBMS. XDB-SQL RDBMS

OS2TREE™

See your entire system graphically at a glance!

"What XTREE GOLD is to DOS and More!

And What Norton Commander Missed"

- Completely programmable: every function/key can be customized!
- Display your tree structure for all system and LAN disks concurrently!
- Manipulate your files, directories, entire disks and/or entire system!
- HPFS fully supported
- Pop-up list of files for any directory
- Edit/browse file(s) using any EDITOR or PROGRAM!
- Extensive FILE Search and Tree directory search capabilities make locating files and/or data easy.
- Upload/download files between mainframe and PC
- Secure delete (sensitive files wiped clean)

Intro Price: \$79.99

Professional Version: \$249.99

Limited time special \$150.00

Network Licenses Available

LEVINE COMPUTER CONSULTING SERVICES

7640 Provincial Dr., Suite 213, McLean, VA 22102

Orders only (800) 383-9495

FAX/Inquiries/HelpLine (703) 790-1660

We'd like to say a word about
client/server development.

TESTING.

You've chosen the platform, architecture, development tools, and application. Now's the time to decide how you're going to test your client/server applications.

The Softbridge Automated Test Facility (ATF) is the *only* product that lets you construct your test cases to replicate realistic client/server scenarios. With ATF, your tests can run the client/server gamut from conditions such as lockout and concurrent requests, to more complex "real world" test cases in which end-user tasks are tightly intertwined.

If you're building client/server (or stand-alone) applications under OS/2 or Windows, you should be using ATF to test them. To learn more about ATF, call 800-955-9190.



Softbridge, Inc.
617-576-2257 (Phone)
617-864-7747 (FAX)

ATF: The final
word in testing.



can be used in a stand-alone environment against a local Windows SQL Engine, as a client workstation accessing any XDB Database Server on a LAN, or connected to mainframe DB2 via XDB-Link. XDB-SQL RDBMS runs under DOS, OS/2, and Windows. Price: Please call.

XDB-SQL C SDK 3.2 comes with a C precompiler and a C-API. XDB's SQL Engine's features include referential integrity, distributed database access, transaction processing, backup and recovery, authorization, concurrency control, and system catalog. C programmers can embed SQL into a program or call SQL engine functions using a powerful C function call library. Applications written using XDB-SQL C SDK can be used either in a stand-alone environment against a resident SQL engine, as a client workstation accessing any XDB-Server on a LAN, or connected to mainframe DB2 via XDB-Link. Price: Please call.

XDB-Server 3.2 is a 32-bit, multiuser database system that features referential integrity, distributed database access,

transaction processing, backup and recovery, authorization, concurrency control, and system catalogue tables. Version 3.3 (available First Quarter, 1994) includes compatibility with DB2 3.0, query optimization enhancements in read-only isolation levels, a governor that allows the user to limit the number of records retrieved, and server-to-server communications. Clients running under Windows NT, Windows, OS/2, and DOS provide user interfaces that gather input and prepare requests for the XDB-Server. The XDB-Server provides storage, access, and control of data. Because the XDB-Server is network-independent, DB2 compatibility can exist on multiple platforms including Windows NT and OS/2. Any of XDB's development, fourth-generation language, and reporting tools, as well as third-party tools including PowerBuilder, can act as clients to the XDB-Server. Price: Please call.

XDB-Link 2.0 contains a set of interface software that links SQL applications on PCs, LANs, and workstations to DB2 data residing on the mainframe.

The XDB-Link product consists of two components: XDB-Link Gateway, which resides on the PC, and a XDB-Link Host, which resides on the mainframe. These components communicate via LU 6.2 protocols to satisfy DB2 SQL requests made from PCs, LANs, and workstations. XDB-Link is based on a client/server architecture. It supports DOS, OS/2, and Windows clients. In a development environment, XDB-Link extends systems level testing to the PC environment by providing access to mainframe production data. In a production environment, XDB's application tools and LAN applications can now access DB2 data residing on the mainframe. XDB-Link Gateway provides direct access to IBM's relational databases, including DB2, SQL/400, and DB2/VM via DRDA. In addition, XDB-Link Host provides direct access to non-relational databases via Remote Procedure Calls (RPCs). Price: Please call.

XDB Systems Inc., 14700 Sweitzer Ln., Laurel, Md. 20702, (301) 317-6800, fax (301) 317-7701.

BUYER'S GUIDE INDEX

Andersen Consulting

(800) 458-8851 or (312) 507-5161

Capstone Software Inc.,

(800) 500-2244 or (317) 848-2414

cfSOFTWARE Inc.,

(800) 366-8756 or (708) 824-7180

Client Server Networking Inc.,

(203) 233-2951

Computer Associates International Inc.,

(800) 225-5224 or (516) 225-5224

Development Technologies Inc.,

(803) 790-9230

Easel Corp.,

(800) 759-5880 or (617) 221-2100

Eicon Technology Corp.,

(514) 631-2592

Gateway Systems Corp.,

(800) 333-9366 or (517) 337-8960

Gilmore Systems,

(805) 379-3210

Glenco Engineering Inc.,

(800) 562-2543 or (708) 808-0300

Intelligent Environments,

(800) 669-2797 or (508) 640-1080

Ipswitch Inc.,

(617) 246-1150

Lane Telecommunications Inc.,

(201) 798-0006

Markham Computer Corp.,

(800) 262-7542 or (407) 394-3994

Mergent International Inc.,

(203) 257-4223

Microformatic,

33(1)48-70-19-00 or (203) 644-1708

Microwork Inc.,

(708) 940-8979

Nettech Systems Inc.,

(609) 683-0100

Open Software Associates,

(603) 886-4330, (408) 730-2626

Parallel PCs Inc.,

(610) 670-1710.

PeerLogic,

(800) 873-7927 or (415) 626-4545

Pinnacle Technology Inc.,

(800) 525-1650 or (317) 279-5157

Quadron Service Corp.,

(805) 966-6424

RightFAX,

(602) 327-1357

SAS Institute Inc.,

(919) 677-8000

S-Cubed Inc.,

(203) 323-0760

Satori Automation

(404) 640-0904

Serengeti Systems Inc.,

(800) 634-3122

Softools Corp.,

(805) 683-5777

Software AG of North America,

(800) 423-2227 or (703) 860-5050

Software Corporation of America,

(800) 966-7722 or (203) 359-2773

The Software Lifeline Inc.,

(407) 994-4466

Strategic Solutions International Corporation,

(203) 649-1900

Szygy Communications Inc.,

(408) 438-5111

Token Technology Inc.,

(415) 965-8607

Uniface Corp.,

(800) 365-3608 or (510) 748-6145

Wall Data Inc.,

(800) 222-0861 or (206) 881-5995

XDB Systems Inc.,

(301) 317-6800

LEARN OBJECTS FAST. DEVELOP 10x FASTER.

BusinessWeek

Want to know the fastest way to learn objects – even for C++ developers? The experts recommend Smalltalk/V, the technology that's 100% pure objects.

FASTER, EASIER LEARNING.

With C++, you have to accomplish two huge tasks: learning object class libraries, and learning new C++ language syntax. Plus C++ is a hybrid of C with added object extensions. So odds are, you'll find yourself constantly falling back on familiar procedural methods and losing the benefit of objects.

Smalltalk/V has a simpler, more approachable language that lets you focus on learning objects instead of a new syntax. That's why thousands of professional programmers have found Smalltalk/V to be the fastest, most efficient way to learn the object-oriented paradigm. In fact,

Smalltalk/V to the rescue. Simply by enabling large programs to be built from pretested software objects, it can provide tenfold leaps in programmer productivity and software quality. Perhaps more important, Smalltalk programs deal in terms programmers can understand.

C++ will make more sense once you learn OOP with Smalltalk/V. But we think you'll want to continue using Smalltalk/V for your application development.

DEVELOP 10x FASTER.

Greg Voss of *Windows Tech Journal* says: "It's not an exaggeration to say that most applications of significant size can be built ten times faster in Smalltalk/V than in C or C++." Gen Kiyooka of *Windows Tech Journal* agrees: "Nothing on earth can match the efficiency and productivity of experienced Smalltalk/V programmers. Nothing." And *Business Week* reminds us that some Wall Street firms now get their "computer models of

"You can work through a Smalltalk/V tutorial in a month and gain a better understanding of OOP than you'll get from six painful months with C++. What you learn will make C++ make sense."

— Greg Voss, OOP Alley

WINDOWS TECH JOURNAL

brand-new financial instruments done in days, not months."

EASY TO USE.

Here's how Smalltalk/V cuts development time by as much as 90%. •Easy-to-use integrated environment. •Incremental compilation shows immediate results of code changes. •Over 350 well-tested reusable classes built-in and 100's more available. •Class browsers and object inspectors to examine code and isolate errors. •Automatic memory management to eliminate memory related bugs. •Easy access to 3GL languages. •Integrated graphic debugger. •Available on Windows 3.1, Windows NT, OS/2 and Macintosh.

So if you want to learn objects fast – and develop 10x faster – punch this line of code into your telephone: (800) 531-2344 Department 706. We'll send you complete information.



SMALLTALK/V. 100% PURE OBJECTS.

DIGITAL



New Borland C++ 4.0 Visual is just the beginning

Why settle for ordinary visual when new Borland C++ gives you so much more? The highest quality fourth-generation tool set. A fully customizable and open desktop. The ability to target

16- and 32-bit Windows simultaneously. And, of course, it's "visual."

Only Borland C++ gives you a fully integrated professional editor featuring BRIEF® technology, powerful Turbo Debugger® GX, and integrated C and C++ VBX control support.

An environment to die for

Borland C++ 4.0 takes the bureaucracy out of development. With the visual Project Manager and its multi-target capabilities, even the most complex projects are handled automatically. The flexibility of AppExpert actually

generates much of your application for you. TargetExpert, ClassExpert, DialogExpert, and Resource Workshop™ streamline the tasks of setting up and customizing your applications.

A true C++ implementation that's years ahead

Languages evolve for a good reason—so programmers can realize ever-increasing productivity and safety of code. That's why Borland is the first to bring important new enhancements to the C++ language, like full support for templates, exception handling, and Run-time Type Information. And Borland C++ includes ObjectWindows™ Library (OWL) 2.0—the world's most popular framework.

Borland C++ is the world-standard C++ because it's the easiest to use, yet has power to spare for the most complex tasks. If you're serious about C++ programming, new Borland C++ 4.0 is the environment you've got to

have. After all, if your compiler is only visual, it's just a facade. Get new Borland C++ 4.0 for Windows and DOS now.



90-day, money-back guarantee!
**See your dealer or call now,
1-800-336-6464, ext. 7158**
In Canada call, 1-800-461-3327.

Borland
Power made easy™
6362-0001-22



Copyright © 1993 Borland International, Inc. All rights reserved. All Borland product names are trademarks of Borland International, Inc.
*Offer good for owners of Microsoft and Symantec C or C++ products; also Turbo C++ owners who purchased product before November 1, 1993. Offer good in the United States and Canada only. All prices in U.S. dollars. Dealer prices may vary. B1 5827

Circle Reader Service Number 52